Interactive and Noninteractive Zero Knowledge are Equivalent in the Help Model*

André Chailloux† Dragos Florin Ciocan‡ Iordanis Kerenidis† Salil Vadhan‡

December 13, 2007

Abstract

We show that interactive and noninteractive zero-knowledge are equivalent in the 'help model' of Ben-Or and Gutfreund (*J. Cryptology*, 2003). In this model, the shared reference string is generated by a probabilistic polynomial-time dealer who is given access to the statement to be proven. Our results do not rely on any unproven complexity assumptions and hold for statistical zero knowledge, for computational zero knowledge restricted to AM, and for quantum zero knowledge when the help is a pure quantum state.

Keywords: cryptography, computational complexity, noninteractive zero-knowledge proofs, commitment schemes, Arthur–Merlin games, quantum zero knowledge

*Preliminary versions of this work previously appeared on the Cryptology ePrint Archive [CK2, CV], and in the second author's undergraduate thesis [Cio].

†LRI, Université Paris-Sud. E-Mail: andre.chailloux@ens-lyon.org, jkeren@lri.fr. Supported in part by ACI Securité Informatique SI/03 511 and ANR AlgoQP grants of the French Ministry and in part by the European Commission under the Integrated Project Qubit Applications (QAP) funded by the IST directorate as Contract Number 015848.

‡School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138. E-Mail: ciocan@post.harvard.edu, salil@eecs.harvard.edu. Supported by NSF Grant CNS-0430336. Some of this work was done when the S. Vadhan was visiting U.C. Berkeley, supported by a Guggenheim Fellowship and the Miller Institute for Basic Research in Science.

1 Introduction

Zero-knowledge proofs [GMR] are protocols whereby a prover can convince a verifier that some assertion is true with the property that the verifier learns nothing else from the protocol. This remarkable property is easily seen to be impossible for the classical notion of a proof system, where the proof is a single string sent from the prover to the verifier, as the proof itself constitutes 'knowledge' that the verifier could not have feasibly generated on its own (assuming NP $\not\subseteq$ BPP). Thus zero-knowledge proofs require some augmentation to the classical model for proof systems.

The original proposal of Goldwasser, Micali, and Rackoff [GMR] augments the classical model with both randomization and multiple rounds of interaction between the prover and the verifier, leading to what are called *interactive zero-knowledge proofs*, or simply *zero-knowledge proofs*. An alternative model, proposed by Blum, Feldman, and Micali [BFM, BDMP], augments the classical model with a set-up in which a trusted dealer randomly generates a *reference string* that is shared between the prover and verifier. After this reference string is generated, the proof consists of a single message from the prover to the verifier. Thus, these are referred to as *noninteractive zero-knowledge proofs*. Since their introduction, there have been many constructions of both interactive and noninteractive zero-knowledge proofs, and both models have found numerous applications in the construction of cryptographic protocols.

It is natural to ask what is the relation between these two models, that is:

> Can every assertion that can be proven with an interactive zero-knowledge proof also be proven with a noninteractive zero-knowledge proof?

Our main result is a positive answer to this question in the 'help model' of Ben-Or and Gutfreund [BG], where the dealer is given access to the statement to be proven when generating the reference string. We hope that this will serve as a step towards answering the above question for more standard models of noninteractive zero knowledge, such as the common reference string model and the public parameter model.

1.1 Models of Zero Knowledge

Interactive Zero Knowledge. Recall that an *interactive proof system* [GMR] for a problem Π is an interactive protocol between a computationally unbounded prover P and a probabilistic polynomial-time verifier V that satisfies the following two properties:

- *Completeness:* if x is a YES instance of Π, then the V will accept with high probability after interacting with the P on common input x.
- *Soundness:* if x is a NO instance of Π, then for every (even computationally unbounded) prover strategy P^*, V will reject with high probability after interacting with P^* on common input x.

Here, we consider problems Π that are not only languages, but also ones that are *promise problems*, meaning that some inputs can be neither YES nor NO instances, and we require nothing of the protocol on such instances. (Put differently, we are 'promised' that the input x is either a YES or a NO instance.) We write IP for the class of promise problems possessing interactive proof systems.

As is common in complexity-theoretic studies of interactive proofs and zero knowledge, we allow the honest prover P to be computationally unbounded, and require soundness to hold against computationally unbounded provers. However, cryptographic applications of zero-knowledge proofs typically require an honest prover P that can be implemented in probabilistic polynomial-time given a witness of membership for x, and it often suffices for soundness to hold only for polynomial-time prover strategies P^* (leading to *interactive argument systems* [BCC]). It was recently shown how to extend the complexity-theoretic studies of interactive zero knowledge proofs to both polynomial-time honest provers [NV], and to argument systems [OV1]; we hope that the same will eventually happen for noninteractive zero knowledge.

Intuitively, we say that an interactive proof system is *zero knowledge* if the verifier 'learns nothing' from the interaction other than the fact that the assertion being proven is true, even if the verifier deviates from the specified protocol. Formally, we require that there is an efficient algorithm, called the *simulator*, that can simulate the verifier's view of the interaction given only the YES instance x and no access to the prover P. The most general notion, *computational zero knowledge* or just *zero knowledge*, requires this to hold for all polynomial-time cheating verifier strategies (and the simulation should be computationally indistinguishable from the verifier's view). A stronger notion, *statistical zero knowledge*, requires security against even computationally unbounded verifier strategies (and the simulation should be statistically indistinguishable from the verifier's view). We write ZK (resp., SZK) to denote the class of promise problems possessing computational (resp., statistical) zero-knowledge proof systems. [1]

Noninteractive Zero Knowledge. For noninteractive zero knowledge [BFM, BDMP], we introduce a trusted third party, the *dealer*, who randomly generates a *reference string* that is provided to both the prover and verifier. After that, the prover sends a single message to the verifier, who decides whether to accept or reject. Completeness and soundness are defined analogously to interactive proofs, except that the probabilities are now also taken over the choice of the reference string. Computational and statistical zero knowledge are also defined analogously to the interactive case, except that now the reference string is also considered part of the verifier's view, and must also be simulated.

There are a number of variants of the noninteractive model, depending on the form of the trusted set-up performed by the dealer. In the original, *common random string (crs) model* proposed by Blum et al. [BFM, BDMP], the reference string is simply a uniformly random string of polynomial length. This gives rise to the classes $\mathrm{NIZK}^{\mathrm{crs}}$ and $\mathrm{NISZK}^{\mathrm{crs}}$ of problems having noninteractive computational and statistical zero-knowledge proofs in the common random string model. A natural and widely used generalization is the *public parameter model*, where the reference string need not be uniform, but can be generated according to any polynomial-time samplable distribution. That is, we obtain the reference string by running a probabilistic polynomial-time *dealer* algorithm D on input 1^n, where n is the length of statements to be proven (or the security parameter). This model gives rise to the classes $\mathrm{NIZK}^{\mathrm{pub}}$ and $\mathrm{NISZK}^{\mathrm{pub}}$.

A further generalization is the *help model* introduced by Ben-Or and Gutfreund [BG]. In this model, the distribution of the reference string is allowed to depend on the statement x being proven. That is, the reference string is generated by running a probabilistic polynomial-time dealer algorithm D on input x. We denote the class of problems having computational (resp. statistical) zero-knowledge proofs in this model as $\mathrm{NIZK}^{\mathrm{h}}$ (resp., $\mathrm{NISZK}^{\mathrm{h}}$). This model does not seem to

[1] In some papers, such as [OV1, OV2], a prefix of C is used to denote *computational* zero knowledge and a suffix of P is used to specify interactive *proof* systems rather than arguments, so ZK and SZK would be CZKP and SZKP, respectively. We opt for more streamlined notation here for readability.

suffice for most cryptographic applications, but its study may serve as a stepping stone towards a better understanding of the more standard models of noninteractive zero knowledge mentioned above. Indeed, any characterizations of noninteractive zero knowledge in the help model already serve as *upper bounds* on the power of noninteractive zero knowledge in the common random string and public parameter models.

We remark that one can also consider protocols in which we allow both a trusted dealer and many rounds of interaction. The most general model allows both help and interaction, yielding the classes ZK^{h} and $\mathrm{SZK}^{\mathrm{h}}$.

Quantum Interactive and Noninteractive Zero Knowledge. The definitions of interactive proofs and zero knowledge extend naturally to the quantum setting. A *quantum interactive proof system* ([KW]) for a promise problem Π is an interactive protocol between a computationally unbounded prover P and a quantum polynomial-time verifier V that satisfies completeness and soundness properties as in the classical case and where the interaction is via quantum messages.

For quantum zero knowledge [Wat1], we require that the verifier's view (which consists of qubits) can be simulated by a quantum polynomial-time machine. QSZK denotes the class of promise problems possessing quantum statistical zero-knowledge proof systems. Kobayashi [Kob] defined quantum noninteractive zero knowledge by having a dealer generate and share a maximally entangled quantum state between the prover and verifier. We write QNISZK to denote the class of promise problems possessing such quantum noninteractive statistical zero-knowledge proof systems.

In this paper, we define two more variants of the quantum noninteractive model, depending on the form of the trusted help created by the dealer. When the help is a *pure* quantum state that depends on the statement x being proven we have the class $\mathrm{QNISZK}^{\mathrm{h}}$. When the help is a *mixed* quantum state that depends on x, we have the class $\mathrm{QNISZK}^{\mathrm{mh}}$. Last, we consider the classes where the dealer creates as help a classical string, which could be either a uniformly random string of polynomial length ($\mathrm{QNISZK}^{\mathrm{crs}}$) or a classical string that depends on the input x ($\mathrm{QNISZK}^{\mathrm{ch}}$). Last, the class $\mathrm{QSZK}^{\mathrm{h}}$ refers to protocols where we allow both a pure quantum help and interaction.

1.2 Previous Work

Recall that we are interested in the relationship between the interactive zero-knowledge classes ZK and SZK and their various noninteractive counterparts, which we will denote by NIZK and NISZK when we do not wish to specify the model. That is, for a given model of noninteractive zero knowledge, we ask: Does ZK = NIZK and SZK = NISZK?

ZK vs. NIZK. A first obstacle to proving equality of ZK and NIZK is that NIZK is a subset of AM, the class of problems having constant-round interactive proof systems [BM, GS], whereas ZK may contain problems outside of AM. So, instead of asking whether ZK = NIZK, we should instead ask if $\mathrm{ZK} \cap \mathrm{AM} = \mathrm{NIZK}$.

Indeed, this equality is known to hold under complexity assumptions. If one-way permutations exist, then it is known that ZK = IP [GMW, IY, BGG+] and $\mathrm{NIZK}^{\mathrm{crs}} = \mathrm{AM}$ [FLS], and thus $\mathrm{ZK} \cap \mathrm{AM} = \mathrm{NIZK}^{\mathrm{crs}} = \mathrm{NIZK}^{\mathrm{pub}} = \mathrm{NIZK}^{\mathrm{h}}$. (In fact, if we replace $\mathrm{NIZK}^{\mathrm{crs}}$ with $\mathrm{NIZK}^{\mathrm{pub}}$, these results hold assuming the existence of any one-way function [HILL, Nao, GB, Pas].) Thus, for computational zero knowledge, the interesting question is whether we can prove that $\mathrm{ZK} \cap \mathrm{AM} =$ NIZK *unconditionally*, without assuming the existence of one-way functions. To our knowledge, there have been no previous results along these lines.

SZK vs. NISZK. For relating SZK and NISZK, the class AM no longer is a barrier, because it is known that SZK $\subseteq$ AM [AH].

The relationship between SZK and NISZK was first addressed in the work of Goldreich et al. [GSV2]. There it was shown that SZK and $\mathrm{NISZK}^{\mathrm{crs}}$ have the 'same complexity' in the sense that SZK = BPP iff $\mathrm{NISZK}^{\mathrm{crs}}$ = BPP. Moreover, it was proven that SZK = $\mathrm{NISZK}^{\mathrm{crs}}$ iff $\mathrm{NISZK}^{\mathrm{crs}}$ is closed under complement.

In addition to introducing the help model, Ben-Or and Gutfreund [BG] studied the relationship between $\mathrm{NISZK}^{\mathrm{h}}$ and SZK. They proved that $\mathrm{NISZK}^{\mathrm{h}} \subseteq$ SZK (in fact that $\mathrm{SZK}^{\mathrm{h}}$ = SZK), and posed as an open question whether SZK $\subseteq \mathrm{NISZK}^{\mathrm{h}}$.[2]

1.3 Our Results

We show that interactive zero knowledge does in fact collapse to noninteractive zero knowledge in the help model, both for the computational case (restricted to AM) and the statistical case:

Theorem 1.1 *$\mathrm{ZK} \cap \mathrm{AM} = \mathrm{NIZK}^{\mathrm{h}}$.*

Theorem 1.2 *$\mathrm{SZK} = \mathrm{NISZK}^{\mathrm{h}}$.*

These results and their proofs yield new characterizations of the classes ZK and SZK. For example, we obtain a new complete problem for SZK, namely the $\mathrm{NISZK}^{\mathrm{h}}$-complete problem given in [BG]. Similarly, we obtain a new characterization of ZK, which amounts to a computational analogue of the $\mathrm{NISZK}^{\mathrm{h}}$-complete problem. As suggested in [BG], these results can also be viewed as first steps towards collapsing interactive zero knowledge to noninteractive zero knowledge in the public parameter or common reference string model. For example, to show SZK = $\mathrm{NISZK}^{\mathrm{crs}}$ (the question posed in [GSV1]), it now suffices to show that $\mathrm{NISZK}^{\mathrm{h}} = \mathrm{NISZK}^{\mathrm{crs}}$.

As mentioned above, one can consider even more general classes ZK^{h} and $\mathrm{SZK}^{\mathrm{h}}$ that incorporate both help and interaction. Ben-Or and Gutfreund [BG] showed that $\mathrm{SZK}^{\mathrm{h}}$ = SZK. We prove an analogous result for computational zero knowledge:

Theorem 1.3 *$\mathrm{ZK}^{\mathrm{h}} = \mathrm{ZK}$.*

In the quantum setting, very little is known about the relation of interactive and noninteractive quantum zero knowledge. Here, we start by providing two complete problems for the class QNISZK. Then, we define two variants of quantum noninteractive zero knowledge depending on the 'help' created by the dealer. In the case where the help is a *pure* quantum state that depends on the input x, we prove an analogue of Theorem 1.2:

Theorem 1.4 *$\mathrm{QNISZK}^{\mathrm{h}} = \mathrm{QSZK} = \mathrm{QSZK}^{\mathrm{h}}$.*

In the case where the help is a *mixed* quantum state, we show that the class $\mathrm{QNISZK}^{\mathrm{mh}}$ contains AM and hence is most probably larger than QSZK. Last, for the quantum noninteractive classes where the help is classical we provide complete problems and show that the message of the prover can always be made classical. This enables us to show that the class $\mathrm{QNISZK}^{\mathrm{ch}}$ is in fact equal to the class of problems that have classical interactive protocols that remain zero knowledge against quantum honest verifiers.

[2] In fact, their conference paper [GB] claimed to prove that SZK = NISZK^{h}, but this was retracted in the journal version [BG].

1.4 Techniques

Here we sketch the techniques underlying the forward inclusions in Theorems 1.1 and 1.2, showing that interactive zero knowledge is a subset of noninteractive zero knowledge in the help model.

We begin with the case of statistical zero knowledge. Our proof that $\mathrm{SZK} \subseteq \mathrm{NISZK}^{\mathrm{h}}$ is similar to the approach suggested by Goldreich *et al.* [GSV2] for showing that $\mathrm{SZK} = \mathrm{NISZK}^{\mathrm{crs}}$. They showed that this question boils down to proving that $\text{co-NISZK}^{\mathrm{crs}} = \mathrm{NISZK}^{\mathrm{crs}}$ or in other words that the complement of the $\mathrm{NISZK}^{\mathrm{crs}}$-complete problem Entropy Approximation belongs to $\mathrm{NISZK}^{\mathrm{crs}}$. Similarly, the core part of our proof is showing that $\text{co-NISZK}^{\mathrm{crs}} \subseteq \mathrm{NISZK}^{\mathrm{h}}$, which then we use to deduce that $\mathrm{SZK} \subseteq \mathrm{NISZK}^{\mathrm{h}}$.

More specifically, our goal is to reduce the SZK-complete problem Entropy Difference (ED) to the NISZK-complete problem Image Intersection Density (IID). Following [GSV2], we start by reducing ED to *several* instances of Entropy Approximation (EA) and its complement ($\overline{\mathrm{EA}}$). We know that $\mathrm{EA} \in \mathrm{NISZK}^{\mathrm{h}}$ since by definition $\mathrm{NISZK}^{\mathrm{crs}} \subseteq \mathrm{NISZK}^{\mathrm{h}}$. Next, inspired by Ben-Or and Gutfreund's attempt [GB] to reduce ED to IID and relying on ideas from [SV1, Oka], we prove that $\overline{\mathrm{EA}}$ also belongs to $\mathrm{NISZK}^{\mathrm{h}}$. Thus we obtain a reduction from ED to several instances of IID. We then conclude our proof by showing that $\mathrm{NISZK}^{\mathrm{h}}$ has enough boolean closure properties to combine these several instances into a *single* instance of IID. We establish these closure properties of $\mathrm{NISZK}^{\mathrm{h}}$ and IID using techniques developed in [SV1, DDPY] to show boolean closure properties for interactive SZK.

In the case of computational zero knowledge, we prove that $\mathrm{ZK} \cap \mathrm{AM} \subseteq \mathrm{NIZK}^{\mathrm{h}}$ by using certain variants of *commitment schemes*. Recall that a commitment scheme is a two-stage interactive protocol between a *sender* and a *receiver*. In the *commit stage*, the sender 'commits' to a secret message m. In the *reveal stage*, the sender 'reveals' m and tries to convince the verifier that it was the message committed to in the first stage. Commitments should be *hiding*, meaning that an adversarial receiver will learn nothing about m in the commit stage, and *binding*, meaning that after the commit stage, an adversarial sender should not be able to successfully reveal two different messages (except with negligible probability). Each of these security properties can be either *computational,* holding against polynomial-time adversaries, or *statistical,* holding even for computationally unbounded adversaries. Commitments are a basic building block for zero-knowledge protocols, e.g. they are the main cryptographic primitive used in the constructions of zero-knowledge proofs for all of NP [GMW] and IP [IY, BGG+].

A relaxed notion is that of *instance-dependent commitment schemes* [BMO, IOS, MV]. Here the sender and receiver are given an instance x of some problem Π as auxiliary input. We only require the scheme to be hiding if x is a YES instance, and only require it to be binding if x is a NO instance. They are a relaxation of standard commitment schemes because we do not require hiding and binding to hold simultaneously. Still, as observed in [IOS], an instance-dependent commitment scheme for a problem $\Pi \in \mathrm{IP}$ suffices to construct zero-knowledge proofs for Π because the constructions of [GMW, IY, BGG+] only use the hiding property for zero knowledge (which is only required on YES instances), and the binding property for soundness (which is only required on NO instances).

We show that a similar phenomenon holds for noninteractive zero knowledge in the help model: If a problem $\Pi \in \mathrm{AM}$ has a certain kind of instance-dependent commitment scheme, then $\Pi \in \mathrm{NIZK}^{\mathrm{h}}$. For this, the instance-dependent commitments naturally need to be *noninteractive*. On the other hand, they only need to be binding (on NO instances) in case the sender is *honest* during the commit phase. (Our observation is that such commitments can be used to implement the hidden bits model of [FLS].)

Thus our task is reduced to showing that every problem in ZK has a noninteractive instance-dependent commitment scheme that is computationally hiding on YES instances and statistically binding for honest senders on NO instances. To prove this, we begin by observing that a problem Π has such an instance-dependent commitment scheme with *statistical* hiding if and only if Π reduces to IID. Hence, the needed commitments already follow for all of SZK from our first result (SZK $\subseteq$ NISZK$^{\mathrm{h}}$). To obtain commitments for all of ZK, we use a characterization of ZK in terms of SZK and 'instance-dependent one-way functions' [Vad], and combine the instance-dependent commitment schemes we obtain from both SZK and the instance-dependent one-way functions.

An alternative construction of the instance-dependent commitments we need can be obtained by using the concurrent work of Ong and Vadhan [OV2]. They showed that every problem in ZK (resp., SZK) has an instance-dependent commitment scheme that is computationally (resp., statistically) hiding on YES instances and statistically binding on NO instances. While their commitments are interactive, they can be made noninteractive if we assume that the sender is honest during the commit phase (by having the sender simulate both parties). Thus, our work can be viewed as a (substantial) simplification to their constructions for the case of honest senders.

2 Definitions and Preliminaries

2.1 Notation

We will first introduce some of the basic notation that we will use.

We use capital letters to denote random variables. The notation $x \leftarrow X$ means that x is drawn from the distribution X. We define the *support* of a random variable X as $\mathrm{Supp}(X) = \{x : \Pr[X = x] > 0\}$. A boolean circuit $C : \{0,1\}^m \to \{0,1\}^n$ defines a probability distribution on $\{0,1\}^n$ by evaluating C on a uniformly chosen input in $\{0,1\}^m$. If a distribution X can be represented by a circuit which can be described and evaluated in polynomial time, we say X is an *efficiently samplable distribution*.

We use the shorthand PPT for probabilistic polynomial time algorithms. For a PPT A, we write $A(x;r)$ to denote the output of A on input x with randomness r. A *nonuniform* PPT algorithm is a pair $(A, \overline{z})$, where $\overline{z}$ is an infinite series of inputs $z_1, \ldots, z_n, \ldots$ such that $|z_n| = \mathrm{poly}(n)$, and A is a PPT which receives inputs $(x, z_{|x|})$.

A function $\varepsilon : N \to [0,1]$ is called *negligible* if $\varepsilon(n) = n^{-\omega(1)}$. We use $\mathrm{neg}(n)$ to denote an arbitrary negligible function, and $\mathrm{poly}(n)$ to denote an arbitrary polynomial function.

2.2 Promise Problems

Promise problems are a more general variant of decision problems than languages. A promise problem Π is a pair of disjoint sets of strings (Π_Y, Π_N), where Π_Y is the set of YES instances and Π_N is the set of NO instances. The computational problem associated with any promise problem Π is: given a string that is "promised" to lie in $\Pi_Y \cup \Pi_N$, decide whether it is in Π_Y or Π_N. Reductions from one promise problem to another are natural extensions of reductions between languages. Namely, we say Π *reduces* to Γ (written $\Pi \preccurlyeq \Gamma$) if there exists a polynomial time computable function f such that $x \in \Pi_Y \Rightarrow f(x) \in \Gamma_Y$ and $x \in \Pi_N \Rightarrow f(x) \in \Gamma_N$. We can also naturally extend the definitions of complexity classes by letting the properties of the strings in the languages be conditions on the YES instances, and properties of strings outside of the language be conditions on NO instances.

2.3 Instance-Dependent Cryptographic Primitives

Many of the objects that we will be constructing for use in our zero knowledge constructions will be instance dependent. Hence, we will modify common cryptographic primitives such as one-way functions by allowing them to be parametrized by some string x, such that the cryptographic properties will only be guaranteed to hold if x is in some set I.

Definition 2.1 *An* instance-dependent function ensemble *is a collection of functions* $\mathcal{F} = \{f_x : \{0,1\}^{p(|x|)} \to \{0,1\}^{q(|x|)}\}_{x \in \{0,1\}^*}$*, where* $p(\cdot)$ *and* $q(\cdot)$ *are polynomials.* $\mathcal{F}$ *is* polynomial-time computable *if there exists a polynomial-time algorithm* F *such that for all* $x \in \{0,1\}^*$ *and* $y \in \{0,1\}^{p(|x|)}$, $F(x,y) = f_x(y)$.

Definition 2.2 *An* instance-dependent one-way function *on* I *is a polynomial-time instance-dependent function ensemble* $\mathcal{F} = \{f_x : \{0,1\}^{p(|x|)} \to \{0,1\}^{q(|x|)}\}_{x \in \{0,1\}^*}$*, such that for every nonuniform PPT* A*, there exists a negligible function* $\varepsilon(\cdot)$ *such that for all* $x \in I$,

$$\Pr\left[A(x, f_x(U_{p(|x|)})) \in f_x^{-1}(f_x(U_{p(|x|)}))\right] \leq \varepsilon(|x|)$$

Definition 2.3 *An* instance-dependent probability ensemble *on* I *is a collection of random variables* $\{X_x\}_{x \in \{0,1\}^*}$*, where* X_x *takes values in* $\{0,1\}^{p(|x|)}$ *for some polynomial* p*. We call such an ensemble* samplable *is there exists a probabilistic polynomial-time algorithm* M *such that for every input* x*,* $M(x)$ *is distributed according to* X_x.

Definition 2.4 *Two instance-dependent probabilistic ensembles* $\{X_x\}$ *and* $\{Y_x\}$ *are* computationally indistinguishable *on* $I \subset \{0,1\}^*$ *if for every nonuniform PPT* D*, there exists a negligible* $\varepsilon(\cdot)$ *such that for all* $x \in I$,

$$\Pr[D(x, X_x) = 1] - \Pr[D(x, Y_x) = 1] \,| \leq \varepsilon(|x|)$$

Similarly, we say $\{X_x\}$ *and* $\{Y_x\}$ *are* statistically indistinguishable *on* $I \subset \{0,1\}^*$ *if the above is required for all functions* D*. If* X_x *and* Y_x *are identically distributed for all* $x \in I$*, we say they are* perfectly indistinguishable .

We will sometimes use the informal notation $X \overset{c}{\equiv} Y$ to denote that ensembles X and Y are computationally indistinguishable.

Definition 2.5 *An* instance-dependent pseudorandom generator *on* I *is a polynomial-time instance-dependent function ensemble* $\mathcal{G} = \{G_x : \{0,1\}^{p(|x|)} \to \{0,1\}^{q(|x|)}\}$ *such that* $q(n) > p(n)$*, and the probability ensembles* $\{G_x(U_{p(|x|)})\}_x$ *and* $\{U_{q(|x|)}\}_x$ *are computationally indistinguishable on* I.

2.4 Probability distributions

In this section, we define several tools that are useful for analyzing properties of probability distributions.

Definition 2.6 *The* statistical difference *between two random variables X and Y taking values in some domain $\mathcal{U}$ is defined as:*

$$\Delta(X,Y) \quad = \quad \max_{S\subset\mathcal{U}} |\Pr[X\in S]-\Pr[Y\in S]| = \frac{1}{2}\sum_{x\in\mathcal{U}} |\Pr[X=x]-\Pr[Y=x]|$$

Definition 2.7 *For an ordered pair of random variables (X,Y), we define their* disjointness *to be:*

$$\mathrm{Disj}(X,Y)=\Pr_X[X\in\mathrm{Supp}(Y)]$$

and we define their mutual disjointness*:*

$$\mathrm{MutDisj}(X,Y)=\min(\mathrm{Disj}(X,Y),\mathrm{Disj}(Y,X)).$$

Note that disjointness is a more stringent measure of the disparity between two distributions than statistical difference. If two distributions have disjointness α, then their statistical difference is at least α. The converse, however, does not hold, since the two distributions could have statistical difference that is negligibly close to 1, yet have identical supports and mutual disjointness 0.

Moreover, we can go from disjoint to mutually-disjoint distributions by the following lemma:

Lemma 2.8 *[BG, SV2] Given a pair of distributions (X_0,X_1) with n input gates, consider the following distributions:*
Y_0: Choose $r\xleftarrow{R}\{0,1\}^n, b\xleftarrow{R}\{0,1\}$, output $(X_b(r),b)$.
Y_1: Choose $r\xleftarrow{R}\{0,1\}^n, b\xleftarrow{R}\{0,1\}$, output $(X_b(r),\overline{b})$.
The following properties hold:

1. $\Delta(Y_0,Y_1)=\Delta(X_0,X_1)$
2. *If (X_0,X_1) is α-disjoint, then (Y_0,Y_1) is mutually $\frac{\alpha}{2}$-disjoint.*

Tensoring Distributions. For random variables X,Y, we let $X\otimes Y$ be the random variable consisting of a sample of X followed by an independent sample of Y. The $\otimes$ notation reflects the fact that the mass function of $X\otimes Y$ is the tensor product of the mass functions of X and Y. When the independence is clear from context, we sometimes write (X,Y) instead of $X\otimes Y$. $X^{\otimes k}$ is the random variable consisting of k independent copies of X.

Lemma 2.9 ([BG, SV2]) *Given a parameter $k\in\mathbb{N}$ and the distributions $X_1,\ldots,X_k$ and $Y_1,\ldots,Y_k$, the pair $(X,Y)=X_1\otimes\ldots\otimes X_k, Y_1\otimes\ldots\otimes Y_k)$ will satisfy the following properties:*

1. $1-2\exp(-k\delta^2/2)\le\Delta(X,Y)\le k\delta$ *where* $\delta=\sum_{i\in[k]}\Delta(X_i,Y_i)/k$.
2. $\mathrm{MutDisj}(X,Y)=1-\prod_{i\in[k]}(1-\alpha_i)$, *where* $\alpha_i=\mathrm{MutDisj}(X_i,Y_i)$.

XORing Distributions. We define the XOR operator which acts on pairs of distributions and returns a pair of distributions. Given two pairs (X_0, X_1) and (X_0', X_1'), with n and n' input gates, respectively, $\mathrm{XOR}((X_0, X_1), (X_0', X_1'))$ is defined by the circuits:

Y_0: Choose $b \xleftarrow{R} \{0,1\}, r \xleftarrow{R} \{0,1\}^n, r' \xleftarrow{R} \{0,1\}^{n'}$, output $(X_b(r), X_b'(r'))$.

Y_1: Choose $b \xleftarrow{R} \{0,1\}, r \xleftarrow{R} \{0,1\}^n, r' \xleftarrow{R} \{0,1\}^{n'}$, output $(X_b(r), X_{\bar{b}}'(r'))$.

Lemma 2.10 (XOR Lemma [BG, SV2]) *If $(Y_0, Y_1) = \mathrm{XOR}((X_0, X_1), (X_0', X_1'))$, then the following properties hold:*

1. $\Delta(Y_0, Y_1) = \Delta(X_0, X_1) \cdot \Delta(X_0', X_1')$.
2. $\mathrm{MutDisj}(Y_0, Y_1) = \mathrm{MutDisj}(X_0, X_1) \cdot \mathrm{MutDisj}(X_0', X_1')$.

By induction, the XOR Lemma implies the following method to decrease both statistical difference and mutual disjointness exponentially fast:

Lemma 2.11 ([BG, SV2]) *Given circuits X_0, X_1 with n input gates and a parameter k, consider the following pair:*

Y_0: Choose $(b_1, \ldots, b_k) \xleftarrow{R} \{(c_1, \ldots, c_k) \in \{0,1\}^k : c_1 \oplus \ldots \oplus c_k = 0\}, (r_1, \ldots r_k) \xleftarrow{R} \{0,1\}^{kn}$, output $(X_{b_1}(r_1), \ldots, X_{b_k}(r_k))$.

Y_1: Choose $(b_1, \ldots, b_k) \xleftarrow{R} \{(c_1, \ldots, c_k) \in \{0,1\}^k : c_1 \oplus \ldots \oplus c_k = 1\}, (r_1, \ldots r_k) \xleftarrow{R} \{0,1\}^{kn}$, output $(X_{b_1}(r_1), \ldots, X_{b_k}(r_k))$.

The following properties hold:

1. $\Delta(Y_0, Y_1) = \Delta(X_0, X_1)^k$.
2. $\mathrm{MutDisj}(Y_0, Y_1) = \mathrm{MutDisj}(X_0, X_1)^k$.

Entropy and Hashing.

Definition 2.12 *The* entropy *of a random variable X is $\mathrm{H}(X) = \mathrm{E}_{x \leftarrow X}\left[\log \frac{1}{\Pr[X=x]}\right]$. The* conditional entropy of X given Y is

$$\mathrm{H}(X|Y) = \mathop{\mathrm{E}}_{y \leftarrow Y}[\mathrm{H}(X|_{Y=y})] = \mathop{\mathrm{E}}_{(x,y) \leftarrow (X,Y)}\left[\log \frac{1}{\Pr[X = x | Y = y]}\right] = \mathrm{H}(X,Y) - \mathrm{H}(Y).$$

For entropy, it holds that for every X, Y, $\mathrm{H}(X \otimes Y) = \mathrm{H}(X) + \mathrm{H}(Y)$. More generally, if $(X,Y)^{\otimes k} = ((X_1, Y_1), \ldots, (X_k, Y_k))$, then $\mathrm{H}((X_1, \ldots, X_k)|(Y_1, \ldots, Y_k) = k \cdot \mathrm{H}(X|Y)$.

Definition 2.13 *The* relative entropy (Kullback-Liebler distance) *between two distributions X, Y is:*

$$\mathrm{KL}(X|Y) = \mathop{\mathrm{E}}_{x \leftarrow X}\left[\log \frac{\Pr[X = x]}{\Pr[Y = x]}\right]$$

We denote by $\mathrm{H}_2(p)$ the binary entropy function, which is the entropy of a $\{0,1\}$-valued random variable with expectation p. $\mathrm{KL}_2(p,q)$ denotes the relative entropy between two $\{0,1\}$-value random variables with expectations p and q.

Flat Distributions. Let X a distribution with entropy $\mathrm{H}(X)$. Elements x of X such that $|\log \Pr[X = x] - \mathrm{H}(X)| \leq k$ are called *k-typical*. We say that X is *Δ-flat* if for every $t > 0$ the probability that an element chosen from X is $t \cdot \Delta$-typical is at least $1 - 2^{-t^2+1}$.

Lemma 2.14 (Flattening Lemma [GV]) *Let X be a distribution encoded by a circuit with n input gates. Then $X^{\otimes k}$ is $\sqrt{k} \cdot n$-flat.*

Definition 2.15 *A family $\mathcal{H}$ of functions from $A \to B$ is* 2-universal *if for every two elements $x \neq y \in A$ and $a, b \in B$, $\Pr_{h \in_R \mathcal{H}}[h(x) = a$ and $h(y) = b] = \frac{1}{|B|^2}$.*

We write $\mathcal{H}_{n,m}$ to denote the 2-universal family from $\{0,1\}^n$ to $\{0,1\}^m$.

Lemma 2.16 (Leftover Hash Lemma [ILL]) *Let $\mathcal{H}$ be a samplable family of 2-universal hashing functions from $A \to B$. Suppose X is a distribution on A such that with probability at least $1 - \delta$ over x selected from X, $\Pr[X = x] \leq \epsilon/|B|$. Consider the following distribution:*

$$Z : \textit{Choose } h \leftarrow \mathcal{H} \textit{ and } x \leftarrow X, \textit{ return } (h, h(x)).$$

Then, $\Delta(Z, \mathcal{U}) \leq O(\delta + \epsilon^{1/3})$, where $\mathcal{U}$ is the uniform distribution on $\mathcal{H} \times B$.

3 Interactive Zero Knowledge

We consider a generalized version of interactive zero knowledge, introduced by Ben-Or and Gutfreund [BG], in which the prover and the verifier have access to a help string output by a dealer algorithm that has access to the statement being proven. We will call this model of interactive zero knowledge the *help model*. Interactive zero-knowledge proofs are a special case of interactive zero-knowledge proofs in the help model.

We denote the three algorithms that make up an interactive zero-knowledge proof in the help model as D, P and V. All three receive as input x, the statement being proven. The dealer selects the help string $\sigma \leftarrow D(x)$ and sends it to P and V. P and V carry out an interactive protocol and, at the end of their interaction, they either output ACCEPT or REJECT. We call the *transcript* the sequence of messages which the triple (D, P, V) computes. $(D, P, V)(x)$ denotes the random variable of the possible outcomes of the protocol, while $\langle D, P, V\rangle(x)$ denotes the verifier's view of the transcripts (where the probability space is over the random coins of D, P and V).

Definition 3.1 ($\mathrm{ZK^h}$, $\mathrm{SZK^h}$ **[BG])** *A* zero-knowledge proof system in the help model *for a promise problem Π is a triple of probabilistic algorithms (D, P, V) (where D and V are polynomial time bounded), satisfying the following conditions:*

1. *Completeness. For all $x \in \Pi_Y$, $\Pr[(D, P, V)(x) = 1] \geq \frac{2}{3}$, where the probability is taken over the coin tosses of D, P and V.*

2. *Soundness. For all $x \in \Pi_N$ and every prover strategy P^*, $\Pr[(D, P^*, V) = 1] \leq \frac{1}{3}$, where the probability is taken over the coin tosses of D, P^*, V.*

3. *Zero Knowledge. There exists a PPT S such that the ensembles $\{\langle D, P, V\rangle)(x)\}_x$ and $\{S(x)\}_x$ are computationally indistinguishable on Π_Y.*

If the ensembles are statistically indistinguishable, we call it a statistical zero knowledge proof system in the help model. ZK^{h} *(resp.,* $\mathrm{SZK}^{\mathrm{h}}$*) is the class of promise problems possessing zero-knowledge (resp., statistical zero-knowledge) proof systems in the help model.*

If the help string σ is generated according to $D(1^{|x|})$, we call the proof system an interactive zero-knowledge proof system in the public parameter model. *The corresponding complexity class is* $\mathrm{ZK}^{\mathrm{pub}}$ *(resp.,* $\mathrm{SZK}^{\mathrm{pub}}$*). If the help string σ is generated from the uniform distribution on $\{0,1\}^{|x|}$, we call the proof system an* interactive zero-knowledge proof system in the common random string model. *The corresponding complexity class is* $\mathrm{ZK}^{\mathrm{crs}}$ *(resp.,* $\mathrm{SZK}^{\mathrm{crs}}$*).*

If we remove the dealer's help, the resulting proof system is said to be an interactive zero-knowledge proof system. *The corresponding complexity class is* ZK *(resp.,* SZK*).*

Note that, in the help model, the dealer is computable in polynomial time given only the instance, and not a witness (hence the notation $D(x)$).

It is simple to show that ZK^{h} is contained in IP, the class of promise problems with interactive proofs:

Lemma 3.2 $\mathrm{ZK}^{\mathrm{h}} \subseteq \mathrm{IP}$.

Proof: We can transform a ZK^{h} proof by just having the verifier simulate the dealer's help. This will not preserve zero knowledge in general, since even the honest verifier will learn the dealer's secret coin tosses, but it will preserve completeness and soundness. ■

3.1 Statistical Zero Knowledge

In this section, we state a few characterizations of statistical zero knowledge which will be related to the ones we will later obtain for the computational case. We begin by noting that, in the statistical case, Ben-Or and Gutfreund [BG] showed that zero knowledge in the help model is equivalent to zero knowledge:

Theorem 3.3 ([BG]) $\mathrm{SZK}^{\mathrm{h}} = \mathrm{SZK}$.

The theorem above implies that all the characterizations of SZK will also hold for $\mathrm{SZK}^{\mathrm{h}}$. In particular, $\mathrm{SZK}^{\mathrm{h}}$ shares the complete problems for SZK that are due to [GV, SV2, Vad]:

Theorem 3.4 ([GV, SV2, Vad]) *The following problems are* SZK*-complete:*

1. Statistical Difference*:*

$$\begin{aligned} \mathrm{SD}_Y &= \{(X,Y) : \Delta(X,Y) \leq 1/3\} \\ \mathrm{SD}_N &= \{(X,Y) : \Delta(X,Y) \geq 2/3\} \end{aligned}$$

where X and Y are samplable distributions specified by circuits that sample from them.

2. Entropy Difference*:*

$$\begin{aligned} \mathrm{ED}_Y &= \{(X,Y) : \mathrm{H}(X) \geq \mathrm{H}(Y)+1\} \\ \mathrm{ED}_N &= \{(X,Y) : \mathrm{H}(Y) \geq \mathrm{H}(X)+1\} \end{aligned}$$

where X and Y are samplable distributions specified by circuits that sample from them.

3. CONDITIONAL ENTROPY APPROXIMATION*:*

$$\begin{aligned} \mathrm{CEA}_Y &= \{(X,Y,r) : H(X|Y) \geq r\} \\ \mathrm{CEA}_N &= \{(X,Y,r) : H(X|Y) \leq r-1\} \end{aligned}$$

where (X,Y) *is a joint samplable distribution specified by circuits that use the same coin tosses.*

Note that we can change the thresholds of 1/3 and 2/3 in SD to other thresholds $\alpha < \beta$. We denote the resulting problem $\mathrm{SD}^{\alpha,\beta}$. It is known that $\mathrm{SD}^{\alpha,\beta}$ is SZK-complete for all constants α, β such that $0 \leq \alpha < \beta^2 \leq 1$ [SV2].

3.2 Computational Zero Knowledge

In the case of ZK, no natural complete problems are known (unless we assume that one-way functions exist, in which case ZK = IP = PSPACE [GMR, IY, BGG+, Sha, LFKN, HILL, Nao]). However, characterizations that are analogous to the complete problems for SZK do exist in the form of the INDISTINGUISHABILITY CONDITION and the CONDITIONAL PSEUDOENTROPY CONDITION below. These conditions give 'if and only if' characterizations of ZK that provide essentially the same functionality that complete problems provide.

The first characterization is a natural computational analogue of STATISTICAL DIFFERENCE:

Definition 3.5 *A promise problem* Π *satisfies the* INDISTINGUISHABILITY CONDITION *if there is a polynomial-time computable function mapping strings* x *to pairs of samplable distributions* (X,Y) *such that:*

- *If* $x \in \Pi_Y$*, then* X *and* Y *are computationally indistinguishable.*
- *If* $x \in \Pi_N$*, then* $\Delta(X,Y) \geq 2/3$*.*

Theorem 3.6 ([Vad]) $\Pi \in \mathrm{ZK}$ *if and only if* $\Pi \in \mathrm{IP}$ *and* Π *satisfies the* INDISTINGUISHABILITY CONDITION*.*

The second characterization is based on the SZK-complete problem CEA:

Definition 3.7 *A promise problem* Π *satisfies the* CONDITIONAL PSEUDOENTROPY CONDITION *if there is a polynomial-time computable function mapping strings* x *to a samplable joint distribution* (X,Y) *such that:*

- *If* $x \in \Pi_Y$*, then there exists a (not necessarily samplable) joint distribution* (X',Y') *such that* (X',Y') *is computationally indistinguishable from* (X,Y) *and* $H(X'|Y') \geq r$*.*
- *If* $x \in \Pi_N$*, then* $H(X|Y) \leq r-1$*.*

Theorem 3.8 ([Vad]) $\Pi \in \mathrm{ZK}$ *if and only if* $\Pi \in \mathrm{IP}$ *and* Π *satisfies the* CONDITIONAL PSEUDOENTROPY CONDITION*.*

Another characterization that we will use is the SZK/OWF CONDITION of [Vad]. The SZK/OWF CONDITION states that any problem in ZK can be decomposed into a part with an SZK proof and another part on which instance-dependent one-way functions can be constructed:

Definition 3.9 (SZK/OWF CONDITION **[Vad]**) *A promise problem* $\Pi = (\Pi_Y, \Pi_N)$ *satisfies the* SZK/OWF CONDITION *if there exists a set* $I \subseteq \Pi_Y$ *of* YES *such that:*

1. *The promise problem* $\Pi' = (\Pi_Y \backslash I, \Pi_N)$ *is in* SZK.
2. *There exists an instance-dependent one-way function on* I *(in the sense of Definition 2.2).*

Theorem 3.10 ([Vad]) $\Pi \in \mathrm{ZK}$ *if and only if* $\Pi \in \mathrm{IP}$ *and* Π *satisfies the* SZK/OWF CONDITION.

4 Noninteractive Zero Knowledge

4.1 The Help Model

In this section, we define the noninteractive analogue of zero-knowledge proofs in the help model.

Definition 4.1 ($\mathrm{NIZK}^{\mathrm{h}}$**,** $\mathrm{NISZK}^{\mathrm{h}}$ **[BG])** *A* noninteractive zero-knowledge proof system in the help model *for a promise problem* Π *is an interactive zero-knowledge proof in which there is only one message* $\pi = P(x, \sigma)$ *from prover to verifier.*

If the real transcripts are statistically indistinguishable from simulated ones, we call it a noninteractive statistical zero knowledge proof system. $\mathrm{NIZK}^{\mathrm{h}}$ *(resp.,* $\mathrm{NISZK}^{\mathrm{h}}$*) is the class of promise problems possessing noninteractive zero-knowledge (resp., noninteractive statistical zero-knowledge) proof systems in the help model.*

If the help string σ *is generated according to* $D(1^{|x|})$*, we call the proof system a* noninteractive zero-knowledge proof system in the public parameter model. *The corresponding complexity class is* $\mathrm{NIZK}^{\mathrm{pub}}$ *(resp.,* $\mathrm{NISZK}^{\mathrm{pub}}$*). If the help string* σ *is generated from the uniform distribution on* $\{0,1\}^{|x|}$*, we call the proof system an* noninteractive zero-knowledge proof system in the common random string model. *The corresponding complexity class is* $\mathrm{NIZK}^{\mathrm{crs}}$ *(resp.,* $\mathrm{NISZK}^{\mathrm{crs}}$*).*

The main benefit of the public parameter model and the help model over the simpler CRS model is that they make it easier to construct NIZK proofs from simpler cryptographic primitives such as one-way functions ([BG, Pas]), or, as we will show in this paper, from noninteractive, instance-dependent commitment schemes.

Like SZK, $\mathrm{NISZK}^{\mathrm{crs}}$ and $\mathrm{NISZK}^{\mathrm{h}}$ exhibit complete problems:

Theorem 4.2 ([GSV2]) *The promise problem* ENTROPY APPROXIMATION*, defined as:*

$$\begin{aligned} \mathrm{EA}_Y &= \{(X,t) : \mathrm{H}(X) \geq t+1\} \\ \mathrm{EA}_N &= \{(X,t) : \mathrm{H}(Y) \leq t-1\} \end{aligned}$$

is complete for $\mathrm{NISZK}^{\mathrm{crs}}$*, where* X *is a samplable distribution specified by a circuit that samples from it. We use the notation* EA^t *to specify an instance of* EA *with parameter* t*.*

Theorem 4.3 ([BG]) *The promise problem* IMAGE INTERSECTION DENSITY*, defined as:*

$$\begin{aligned}\mathrm{IID}_Y &= \{(X,Y) : \Delta(X,Y) \leq 1/3\} \\ \mathrm{IID}_N &= \{(X,Y) : \mathrm{MutDisj}(X,Y) \geq 2/3\}\end{aligned}$$

is complete for $\mathrm{NISZK}^{\mathrm{h}}$*, where* X *and* Y *are samplable distributions specified by circuits that sample from them.*

We note that our definition of IID is slightly different than the one used by [BG]. In our definition, we are working with mutual disjointness, since it is easy to transform disjoint distributions to mutually disjoint ones (Lemma 2.8). Additionally, due to a stronger Polarization Lemma that we will describe in a subsequent section, we use constant thresholds of 1/3 and 2/3 rather than functions tending to 0 and 1.

We also recall the complexity class AM, which is is the class of promise problems possessing constant-round interactive proofs, or equivalently, 2-round public-coin interactive proofs [BM, GS]:

Definition 4.4 (AM) *An* AM proof system *is a pair of probabilistic algorithms* (P, V) *where the prover* P *(sometimes called Merlin) is unbounded, whereas the verifier* V *(sometimes called Arthur) is PPT.* V *sends a random string* $r \xleftarrow{R} \{0,1\}^{\mathrm{poly}(|x|)}$*, to which* P *sends a single response* m*.* V *decides then accepts or rejects with no more randomness (i.e.* V *is a deterministic function of* x, r *and* m*). Equivalently, a promise problem* $\Pi \in \mathrm{AM}$ *if* $\exists$ *a polynomial-time algorithm* V*, and polynomials* $p(|x|), q(|x|)$ *such that:*

1. *Completeness.* $x \in \Pi_Y \Rightarrow \Pr_{r \in \{0,1\}^{p(|x|)}}[\exists m \in \{0,1\}^{q(|x|)} s.t.\ V(x,r,m) = 1] \geq 2/3.$
2. *Soundness.* $x \in \Pi_N \Rightarrow \Pr_{r \in \{0,1\}^{p(|x|)}}[\exists m \in \{0,1\}^{q(|x|)} s.t.\ V(x,r,m) = 1] \leq 1/3.$

Analogous to Lemma 3.2, AM proves to be a natural upper bound for $\mathrm{NIZK}^{\mathrm{h}}$, since we can just have the verifier replace the dealer in creating the reference string. Also, a lower bound for $\mathrm{NIZK}^{\mathrm{h}}$ is $\mathrm{NIZK}^{\mathrm{crs}}$, which is definitionally a more restricted version of the help model.

5 Quantum preliminaries and definitions

5.1 The quantum formalism

Let $\mathcal{H}$ denote a 2-dimensional complex vector space, equipped with the standard inner product. We pick an orthonormal basis for this space, label the two basis vectors $|0\rangle$ and $|1\rangle$. , and for simplicity identify them with the vectors $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$, respectively. A *qubit* is a unit length vector in this space, and so can be expressed as a linear combination of the basis states: $\alpha_0|0\rangle + \alpha_1|1\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}$. Here α_0, α_1 are complex *amplitudes*, and $|\alpha_0|^2 + |\alpha_1|^2 = 1$.

An *m-qubit pure state* is a unit vector in the m-fold tensor space $H \otimes \cdots \otimes H$. The 2^m basis states of this space are the m-fold tensor products of the states $|0\rangle$ and $|1\rangle$. For example, the basis states of a 2-qubit system are the four 4-dimensional unit vectors $|0\rangle \otimes |0\rangle$, $|0\rangle \otimes |1\rangle$, $|1\rangle \otimes |0\rangle$, and

$|1\rangle \otimes |1\rangle$. We abbreviate, e.g., $|1\rangle \otimes |0\rangle$ to $|0\rangle|1\rangle$, or $|1,0\rangle$, or $|10\rangle$, or even $|2\rangle$ (since 2 is 10 in binary). With these basis states, an m-qubit state $|\phi\rangle$ is a 2^m-dimensional complex unit vector

$$|\phi\rangle = \sum_{i \in \{0,1\}^m} \alpha_i |i\rangle.$$

We use $\langle\phi| = |\phi\rangle^*$ to denote the conjugate transpose of the vector $|\phi\rangle$, and $\langle\phi|\psi\rangle = \langle\phi| \cdot |\psi\rangle$ for the inner product between states $|\phi\rangle$ and $|\psi\rangle$. These two states are *orthogonal* if $\langle\phi|\psi\rangle = 0$. The *norm* of $|\phi\rangle$ is $\| \phi \| = \sqrt{\langle\phi|\phi\rangle}$.

A *mixed state* $\{p_i, |\phi_i\rangle\}$ is a classical distribution over pure quantum states, where the system is in state $|\phi_i\rangle$ with probability p_i. We can represent a mixed quantum state by the *density matrix* which is defined as $\rho = \sum_i p_i |\phi_i\rangle\langle\phi_i|$. Note that ρ is a positive semidefinite operator with trace (sum of diagonal entries) equal to 1. The density matrix of a pure state $|\phi\rangle$ is $\rho = |\phi\rangle\langle\phi|$.

A quantum system is called *bipartite* if it consists of two subsystems. We can describe the state of each of these subsystems separately with the *reduced density matrix*. For example, if the joint quantum state of two subsystems A, B has the form $|\phi\rangle = \sum_i \sqrt{p_i}|i\rangle_A|\phi_i\rangle_B$, then the state of the subsystem B, *i.e.*, the subsystem which contains only the second part of $|\phi\rangle$ is described by the (reduced) density matrix $\sum_i p_i|\phi_i\rangle\langle\phi_i|$.

A quantum state evolves by a unitary operation or by a measurement. A *unitary* transformation U is a linear mapping that preserves the complex ℓ_2 norm. If we apply U to a state $|\phi\rangle$, it evolves to $U|\phi\rangle$. A mixed state ρ evolves to $U\rho U^\dagger$.

The most general measurement allowed by quantum mechanics is specified by a family of positive semidefinite operators $E_i = M_i^* M_i$, $1 \leq i \leq k$, subject to the condition that $\sum_i E_i = I$. Given a density matrix ρ, the probability of observing the ith outcome under this measurement is given by the trace $p_i = \mathrm{Tr}(E_i\rho) = \mathrm{Tr}(M_i\rho M_i^*)$. These p_i are nonnegative because E_i and ρ are positive semidefinite. They also sum to 1, as they should:

$$\sum_{i=1}^{k} p_i = \sum_{i=1}^{k} \mathrm{Tr}(E_i\rho) = \mathrm{Tr}\left(\sum_{i=1}^{k} E_i\rho\right) = \mathrm{Tr}(I\rho) = 1.$$

If the measurement yields outcome i, then the resulting mixed quantum state is $M_i\rho M_i^*/\mathrm{Tr}(M_i\rho M_i^*)$. In particular, if $\rho = |\phi\rangle\langle\phi|$, then $p_i = \langle\phi|E_i|\phi\rangle = \| M_i|\phi\rangle \|^2$, and the resulting state is $M_i|\phi\rangle/\| M_i|\phi\rangle \|$. A special case is where $k = 2^m$ and $B = \{|\psi_i\rangle\}$ forms an orthonormal basis of the m-qubit space. 'Measuring in the B-basis' means that we apply the measurement given by $E_i = M_i = |\psi_i\rangle\langle\psi_i|$. Applying this to a pure state $|\phi\rangle$ gives resulting state $|\psi_i\rangle$ with probability $p_i = |\langle\phi|\psi_i\rangle|^2$.

The trace norm of a matrix A is denoted by $||A||$ and is equal to the trace of $|A|$, where $|A| = \sqrt{A^\dagger A}$ is the positive square root of $A^\dagger A$. For two density matrices ρ_1, ρ_2 we define their trace distance as the trace norm of the matrix $\rho_1 - \rho_2$, *i.e.*, $||\rho_1 - \rho_2||$.

The von Neumann Entropy of a mixed quantum state ρ with eigenvalues λ_i is defined as $S(\rho) = -\sum_i \lambda_i \log \lambda_i$.

5.2 Quantum Interactive and Noninteractive Statistical Zero-Knowledge

Quantum statistical zero knowledge proofs are a special case of quantum interactive proofs. We can think of a *quantum interactive protocol* $\langle P, V\rangle(x)$ as a series of circuits $(V_1(x), P_1(x), \ldots, V_k(x), P_k(x))$ on the space $\mathcal{V} \otimes \mathcal{M} \otimes \mathcal{P}$. $\mathcal{V}$ are the verifier's private qubits, $\mathcal{M}$ are the message qubits and $\mathcal{P}$

are the prover's private qubits. $V_i(x)$ (resp. $P_i(x)$) represents the i^{th} action of the verifier (resp. the prover) during the protocol and acts on $\mathcal{V} \otimes \mathcal{M}$ (resp. $\mathcal{M} \otimes \mathcal{P}$). β_i corresponds to the state that appears after the i^{th} action of the protocol. We define completeness and soundness exactly the same way as in the case of classical protocols. We say that a protocol $\langle P, V\rangle$ *solves* Π if it has completeness greater than 2/3 and soundness less than 1/3.

In the zero knowledge setting, we also want that the verifier learns nothing from the interaction other than the fact that $x \in \Pi_Y$ when it is the case. The way it is formalized is that for $x \in \Pi_Y$, the verifier can simulate his view of the protocol. We are interested only in honest verifier protocols where the verifier and the prover use unitary operations, since by Watrous [Wat2] we know that honest verifier with unitary operations is equivalent to cheating verifier (that is allowed to use any permissible operation).

Let $\langle P, V\rangle$ a quantum protocol and β_j defined as before. The verifier's *view* of the protocol is his private qubits and the message qubits, $\text{view}_{\langle P,V\rangle}(j) = Tr_{\mathcal{P}}(\beta_j)$. We also want to separate the verifier's view based on whether the last action was made by the verifier or the prover. We note ρ_0 the input state, ρ_i the verifier's view of the protocol after P_i and ξ_i the verifier's view of the protocol after V_i.

Definition 5.1 *A quantum protocol $\langle P, V\rangle$ has the zero knowledge property for Π if there exists a quantum polynomial-time simulator σ and a negligible function μ such that for every input $x \in \Pi_Y$ and $\forall j$ $\|\sigma_j(x) - \rho_j\| \leq \mu(|x|)$.*

Note that for a state σ such that $\|\sigma - \rho_i\| \leq \mu(|x|)$ it is easy to see that $\sigma' = V_{i+1}\sigma V_{i+1}^{\dagger}$ is close to $\xi_{i+1} = V_{i+1}\rho_i V_{i+1}^{\dagger}$ in this sense that $\|\sigma' - \xi_{i+1}\| \leq \mu(|x|)$. Therefore, in the definition we just need to simulate the ρ_i's. Also note that the simulation in the quantum case is done round by round which seems to be a weaker definition than in the classical case. However, since the message qubits are reused in every round, the notion of a transcript can not be defined in the quantum case.

Definition 5.2 $\Pi \in QSZK$ *iff there exists a quantum protocol $\langle P, V\rangle$ that solves Π and that has the zero-knowledge property for Π.*

In the setting of quantum noninteractive statistical zero knowledge, first defined by Kobayashi [Kob], the prover and verifier share a maximally entangled state $\sum_i |i\rangle_P |i\rangle_V$ created by a trusted third party: the dealer D. Then the prover sends a single quantum message to the verifier. We can assume that the message from the dealer to the verifier goes into his private space $\mathcal{V}$. Hence, after the prover's message, the verifier's view ρ_1 also contains the message from the dealer.

In this setting, we define the zero knowledge property as follows:

Definition 5.3 *A quantum noninteractive protocol $\langle D, P, V\rangle$ has the zero know-ledge property for Π if there exists a quantum polynomial-time simulator σ and a negligible function μ such that for every input $x \in \Pi_Y$ $\|\sigma(x) - \rho_1\| \leq \mu(|x|)$.*

Definition 5.4 $\Pi \in$ QNISZK *iff, when the prover and verifier share the maximally entangled state $\sum_i |i\rangle_P |i\rangle_V$ created by the dealer D, there exists a quantum noninteractive protocol $\langle D, P, V\rangle$ that solves Π and that has the zero-knowledge property for Π.*

The quantum analogues of the classical complete problems can be easily defined in the following way: the inputs are now quantum mixed states computable in polynomial time and the distance measures are the trace distance (instead of the statistical distance) and the von Neumann Entropy (instead of the Shannon Entropy).

For example, we denote by QUANTUM STATE DISTINGUISHABILITY (QSD) the following promise problem :

$$\begin{aligned} \mathrm{QSD}_Y &= \{(\rho_1,\rho_2) : ||\rho_1-\rho_2|| \le 1/3\} \\ \mathrm{QSD}_N &= \{(\rho_1,\rho_2) : ||\rho_1-\rho_2|| \ge 2/3\} \end{aligned}$$

where ρ_1 and ρ_2 are quantum mixed states which can be created in polynomial time using a quantum computer.

6 Statistical Zero Knowledge

6.1 The Polarization Lemma

Zero knowledge protocols usually require from promise problems some parameters that are exponentially close to 0 or 1. Polarizations are reductions from promise problems with weak parameters to promise problems that can be solved by the protocols. For example, there is a polarization for the promise problem SD that transforms $\mathrm{SD}^{a,b}$ with $a^2 > b$ to $\mathrm{SD}^{1-2^{-k},2^{-k}}$ for any $k = \mathrm{poly}(n)$ [SV2].

The best polarization that was known for IID was that $\mathrm{IID}^{1/n^2,1-1/n^2}$ reduces to $\mathrm{IID}^{2^{-k},1-2^{-k}}$ and henceforth $\mathrm{IID}^{1/n^2,1-1/n^2}$ is complete for $\mathrm{NISZK}^{\mathrm{h}}$ [BG]. We will show here that $\mathrm{IID}^{a,b}$ is complete for $\mathrm{NISZK}^{\mathrm{h}}$ with $b > a$ (where a and b are constants).

Lemma 6.1 (Polarization Lemma [BG, SV2]) *There exists an algorithm that takes a pair of distributions (X_0, X_1) and parameters $n \in \mathbb{N}, 0 \le \alpha < \beta \le 1$, and outputs a pair of distributions (Y_0, Y_1) such that:*

1. $\Delta(X_0, X_1) \le \alpha \Rightarrow \Delta(X_0, X_1) \le 2^{-n}$.
2. $\mathrm{MutDisj}(X_0, X_1) \ge \beta \Rightarrow \mathrm{MutDisj}(Y_0, Y_1) \ge 1 - 2^{-n}$.

The algorithm runs in time $\mathrm{poly}\left(|(X_0, X_1)|, n, \exp\left(\frac{\alpha \log(1/\beta)}{\beta-\alpha}\right)\right)$.

Proof: Let $\lambda = \min\{\beta/\alpha, 2\} > 1$.

We first apply Lemma 2.11 with $k = \log_\lambda 2n$, obtaining two distributions which are either statistically α^k close, or have β^k mutual disjointness.

Then, we apply Lemma 2.9 with $m = \lambda^k/(2\beta^k) \le 1/(2\alpha^k)$. This gives two distributions with either statistical difference at most $m\alpha^k \le 1/2$, or mutual disjointness of at most $1-(1-\beta^k)^m \ge 1 - e^{-\beta^k m} = 1 - e^{-\beta^k \cdot \lambda^k/(2\beta^k)} = 1 - e^{-\lambda^k/2} = 1 - e^{-n}$.

Finally, we apply again Lemma 2.11 with parameter n to get either statistical difference at most 2^{-n}, or mutual disjointness at most $(1-e^{-n})^n \ge 1 - ne^{-n} \ge 1 - 2^{-n}$, for sufficiently large n.

The running time of the algorithm is $\mathrm{poly}(|(X_0, X_1)|, n, k)$, where $k = O(\log n/(\lambda - 1)) = O(\alpha \log n/(\beta - \alpha))$ and $m \le 1/2 \cdot (2/\beta)^k = \exp\left(O\left(\frac{\alpha \log n \log(2/\beta)}{\beta-\alpha}\right)\right)$. This gives the claimed running time if either $n = O(1)$ or if $\beta - \alpha = \Omega(1)$. Thus we can obtain the lemma by applying the transformation in two steps, first with $n' = 2$ to polarize to thresholds $\alpha' = 1/4$ and $\beta' = 3/4$, and then once more with the desired value of n. ■

This can be compared to the original Polarization Lemma of [SV2], which refers to statistical difference in Item 2 (rather than mutual disjointness), but only achieves polarization from thresholds such that $0 \leq \alpha < \beta^2 \leq 1$, and for which it is known that the gap between thresholds is inherent for a natural class of transformations [HR].

We also add that, at a factor of 2 in β, we can start with β-disjoint distributions rather than mutually β-disjoint ones for the polarization to work. The reason is that we can easily transform a pair (X, Y) that is 2β-disjoint into a pair (X', Y') such that $\Delta(X', Y') = \Delta(X, Y)$ and (X', Y') is mutually β-disjoint, using Lemma 2.8.

6.2 SZK and NISZK$^{\mathrm{h}}$ Are Equivalent

We show in this section that help and interaction are equivalent in the statistical zero knowledge setting.

Theorem 6.2 $\mathrm{SZK} = \mathrm{NISZK}^{\mathrm{h}}$

The inclusion $\mathrm{NISZK}^{\mathrm{h}} \subseteq \mathrm{SZK}$ was proven by Ben-Or and Gutfreund [BG], since the $\mathrm{NISZK}^{\mathrm{h}}$-complete problem Image Intersection Density (IID) trivially reduces to Statistical Difference (SD), the SZK-complete problem. In what follows, we prove the opposite inclusion by reducing the SZK-complete problem Entropy Difference (ED) to IID. Ben-Or and Gutfreund claimed to have proven this reduction in [GB] but due to a flaw they retracted it in [BG]. Their reduction from ED to IID was in fact only a reduction to SD. Still, part of our proof is inspired by their method.

In order to prove that $\mathrm{SZK} \subseteq \mathrm{NISZK}^{\mathrm{h}}$, we follow [GSV2] and reduce the SZK-complete problem ED to several instances of Entropy Approximation and its complement (EA and $\overline{\mathrm{EA}}$) using the following fact:

Fact 6.3 ([GSV2]) *Let $X' = X^{\otimes 3}$ and $Y' = Y^{\otimes 3}$. Let n the output size of X' and Y'. It holds that:*

$$(X, Y) \in \mathrm{ED}_Y \Leftrightarrow \forall t \in \{1, \ldots, n\} \left[((X', t) \in \mathrm{EA}_Y) \vee ((Y', t) \in \overline{\mathrm{EA}}_Y)\right]$$
$$(X, Y) \in \mathrm{ED}_N \Leftrightarrow \exists t \in \{1, \ldots, n\} \left[((X', t) \in \mathrm{EA}_N) \wedge ((Y', t) \in \overline{\mathrm{EA}}_N)\right]$$

We know that $\mathrm{EA} \in \mathrm{NISZK}^{\mathrm{h}}$ (since by definition $\mathrm{NISZK}^{\mathrm{crs}} \subseteq \mathrm{NISZK}^{\mathrm{h}}$), so it remains to show the following two things:

1. $\overline{\mathrm{EA}} \in \mathrm{NISZK}^{\mathrm{h}}$: in order to this, we reduce $\overline{\mathrm{EA}}$ to IID, inspired by Ben-Or and Gutfreund's attempt [GB] to reduce ED to IID. This reduction relies on ideas from [SV1, Oka].

2. $\mathrm{NISZK}^{\mathrm{h}}$ has certain boolean closure properties: this will allow us to reduce ED to a single instance of IID. Since IID and SD are closely related, we use similar techniques to the ones used in [SV1, DDPY].

Note that our proof's structure is similar to the approach suggested by Goldreich *et al.* [GSV2] for showing that $\mathrm{NISZK}^{\mathrm{crs}} = \mathrm{SZK}$. They proved that if $\mathrm{NISZK}^{\mathrm{crs}} = \text{co-}\mathrm{NISZK}^{\mathrm{crs}}$ then $\mathrm{NISZK}^{\mathrm{crs}} = \mathrm{SZK}$. We show here that $\text{co-}\mathrm{NISZK}^{\mathrm{crs}} \subseteq \mathrm{NISZK}^{\mathrm{h}}$, and using the closure properties, conclude that $\mathrm{NISZK}^{\mathrm{h}} = \mathrm{SZK}$.

6.3 $\overline{\text{EA}}$ belongs to NISZK^{h}

In this section, we prove the following lemma:

Lemma 6.4 $\overline{\text{EA}} \in \text{NISZK}^{\text{h}}$.

Proof: We will reduce $\overline{\text{EA}}$ to IID, which is complete for NISZK^{h}.

Let (X,t) an instance of $\overline{\text{EA}}$. By artificially adding input gates or output gates to X, we can assume that X has m input and output gates. Let k a large constant that will be specified later on and $X' = X^{\otimes s}$ with $s = 4km^2$. Note that X' has $m' = s \cdot m$ input and output gates and $\text{H}(X') = s \cdot \text{H}(X)$. We have:

Fact 6.5

1. *X' is Δ-flat with $\Delta = 2\sqrt{k}m^2$, where s was chosen such that $s = 2\sqrt{k}\Delta$.*
2. $\Pr[X'$ *is $\sqrt{k}\Delta$-typical* $] \geq 1 - 2^{-\Omega(k)}$.

Given (X,t), we can create two distributions Z as Z' as following

Z: Choose $r \overset{R}{\leftarrow} \{0,1\}^{m'}$, $x = X'(r)$, $h \overset{R}{\leftarrow} \mathcal{H}_{m'+st,m'}$, $z \overset{R}{\leftarrow} \{0,1\}^{m'}$. Return (x,h,z).
Z': Choose $r \overset{R}{\leftarrow} \{0,1\}^{m'}$, $x = X'(r)$, $h \overset{R}{\leftarrow} \mathcal{H}_{m'+st,m'}$, $u \overset{R}{\leftarrow} \{0,1\}^{st}$. Return $(x,(h,h(r,u)))$.

Note that Z' is of the form $Z' = (X', A)$. We write A_x to denote the distribution of A conditioned on $X' = x$. Note that we can describe A_x as follows :
A_x : Choose $r \overset{R}{\leftarrow} (X')^{-1}(x)$, $h \overset{R}{\leftarrow} \mathcal{H}_{m'+st,m'}$, $u \overset{R}{\leftarrow} \{0,1\}^{st}$ and return $(h,h(r,u))$.
Hence, we need to show that, when conditioning on $X' = x$, we have either $\Delta(\mathcal{U}, A_x)$ small (on the YES instances) or $\text{Disj}(\mathcal{U}, A_x)$ large (on the NO instances).

For $x \in \text{Supp}(X')$, let $\text{wt}(x) = \log|(X')^{-1}(x)| = m' - \log(\frac{1}{Pr[X'=x]})$. The number of different possible inputs (r,u) that are hashed in A_x is $2^{\text{wt}(x)+st}$. Using Fact 6.5, it is easy to see that, if $\text{H}(X) \leq t-1$, then $\text{wt}(x)$ will be large with high probability, whereas, if $\text{H}(X) \geq t+1$, then $\text{wt}(x)$ will be small with high probability. We can now show the following two claims which will allow us to conclude the proof.

Claim 6.6 $(X,t) \in \overline{\text{EA}}_Y \Rightarrow \Delta(Z,Z') = 2^{-\Omega(k)}$.

Proof: For all $x \in \text{Supp}(X')$ that are $\sqrt{k}\Delta$-typical, $\left|\log(\frac{1}{Pr[X'=x]}) - \text{H}(X')\right| \leq \sqrt{k}\Delta$. Hence,

$$\text{wt}(x) \geq m' - s \cdot \text{H}(X) - \sqrt{k}\Delta \geq m' - st + s - \sqrt{k}\Delta \geq m' - st + \sqrt{k}\Delta.$$

Therefore, the number of inputs (r,u) such that $X'(r) = x$ and $u \in \{0,1\}^{st}$ is greater than $2^{m'+\sqrt{k}\Delta} \geq 2^{m'+k}$. By the Leftover Hash Lemma (Lemma 2.16), $\Delta(\mathcal{U}, A_x) = 2^{-\Omega(k)}$. By Fact 6.5, the probability of a $\sqrt{k}\Delta$-typical x is $1 - 2^{-\Omega(k)}$ and hence we can conclude that $\Delta(Z,Z') = 2^{-\Omega(k)}$. ■

Claim 6.7 $(X,t) \in \overline{\text{EA}}_N \Rightarrow \text{Disj}(Z,Z') = 1 - 2^{-\Omega(k)}$.

Proof: For all $x \in \text{Supp}(X')$ that are $\sqrt{k}\Delta$-typical, we have:

$$\text{wt}(x) \le m' - s \cdot \text{H}(X) + \sqrt{k}\Delta \le m' - st - s + \sqrt{k}\Delta \le m' - st - \sqrt{k}\Delta.$$

Therefore, the number of inputs (r, u) such that $X'(r) = x$ and $u \in \{0,1\}^{st}$ is smaller than $2^{m'-\sqrt{k}\Delta} \le 2^{m'-k}$. Since we hash at most $2^{m'-k}$ values into $\{0,1\}^{m'}$, we get only a 2^{-k} fraction of the total support and hence $\text{Disj}(\mathcal{U}, A_x) = 1 - 2^{-\Omega(k)}$. By Fact 6.5, the probability of a $\sqrt{k}\Delta$-typical x is $1 - 2^{-\Omega(k)}$ and hence we can conclude that $\text{Disj}(Z, Z') = 1 - 2^{-\Omega(k)}$.

■

By taking k a large enough constant, we can ensure that $(X,t) \in \overline{\text{EA}}_Y \Rightarrow \Delta(Z, Z') \le 1/4$ and also $(X,t) \in \overline{\text{EA}}_N \Rightarrow \text{Disj}(Z, Z') \ge 3/4$.

The only thing that remains is to transform the disjointness in the NO instances to mutual disjointness. We first apply Lemma 2.8 to create distributions (A, B) such that $\Delta(A, B) \le 1/4$ or $\text{Disj}(A, B) \ge 3/8$. Then, by the polarization Lemma shown in Subsection 6.1, we create distributions (A', B') such that $(X,t) \in \overline{\text{EA}}_Y \Rightarrow \Delta(A', B') \le 1/3$ and $(X,t) \in \overline{\text{EA}}_N \Rightarrow \text{Disj}(A', B') \ge 2/3$.

In conclusion, we see that from (X,t), we have created distributions A', B' in polynomial time such that :

- $(X,t) \in \overline{\text{EA}}_Y \Rightarrow (A', B') \in \text{IID}_Y$.
- $(X,t) \in \overline{\text{EA}}_N \Rightarrow (A', B') \in \text{IID}_N$.

Hence, $\overline{\text{EA}}$ reduces to IID and from the completeness of IID for NISZK^{h}, we have $\overline{\text{EA}} \in \text{NISZK}^{\text{h}}$.

■

6.4 Closure properties for NISZK^{h}

We now prove some closure properties of NISZK^{h} that we will use to complete the proof of Theorem 6.2. Every promise problem $\Pi \in \text{NISZK}^{\text{h}}$ reduces to IID and hence, we just have to concentrate on this problem. Note that this problem is very similar to the SZK-complete promise problem SD and hence we use similar techniques to those developed in [DDPY, SV1] to show closure properties for SZK. In our case, we just need to show some limited closure properties that will be enough to prove that $\overline{\text{ED}} \in \text{NISZK}^{\text{h}}$.

Definition 6.8 *Let* Π *some promise problem. We define* $\text{AND}(\Pi)$ *to be the following promise problem:*

- $\text{AND}(\Pi)_Y = \{(x_1, \ldots, x_k) : \ \forall i \in \{1, \ldots, k\} \ x^i \in \Pi_Y\}$.
- $\text{AND}(\Pi)_N = \{(x_1, \ldots, x_k) : \ \exists i \in \{1, \ldots, k\} \ x^i \in \Pi_N\}$.

Similarly, we define $\text{OR}(\Pi)$ for a pair of instances of Π.

Definition 6.9 *Let* Π *a promise problem. We define* $\text{OR}(\Pi)$ *to be the following promise problem:*

- $\text{OR}(\Pi)_Y = \{(x_1, x_2) : \ \exists i \in \{1, 2\} \ x^i \in \Pi_Y\}$.
- $\text{OR}(\Pi)_N = \{(x_1, x_2) : \ \forall i \in \{1, 2\} \ x^i \in \Pi_N\}$.

We show that $\mathrm{NISZK^h}$ is closed under AND and OR.

Lemma 6.10 $\mathrm{NISZK^h}$ *is closed under* AND.

Proof: Let Π be in $\mathrm{NISZK^h}$ and $(x_1, \ldots, x_k)$ be an instance of AND(Π). We reduce Π to the IID problem which means that we transform each x_i into a pair of distributions (X^i, Y^i) such that $x_i \in \Pi_Y \Rightarrow (X^i, Y^i) \in \mathrm{IID}_Y$ and $x_i \in \Pi_N \Rightarrow (X^i, Y^i) \in \mathrm{IID}_N$. Let $X = X^1 \otimes \cdots \otimes X^k$ and $Y = Y^1 \otimes \cdots \otimes Y^k$. We first polarize each pair (X^i, Y^i) to have statistical difference at most $1/3k$ or mutual disjointness at least $2/3$. From Lemma 2.9, we can easily see that $(x_1, \ldots, x_k) \in \mathrm{AND}(\Pi)_Y \Rightarrow (X, Y) \in \mathrm{IID}_Y$ and that $(x_1, \ldots, x_k) \in \mathrm{AND}(\Pi)_N \Rightarrow (X, Y) \in \mathrm{IID}_N$, which concludes our proof. ■

Lemma 6.11 $\mathrm{NISZK^h}$ *is closed under* OR.

Proof: Let Π be in $\mathrm{NISZK^h}$. Let (x_1, x_2) be an instance of OR(Π). We reduce Π to the IID problem which means that we transform each x_i into a pair of distributions (X^i, Y^i) such that $x_i \in \Pi_Y \Rightarrow (X^i, Y^i) \in \mathrm{IID}_Y$ and $x_i \in \Pi_N \Rightarrow (X^i, Y^i) \in \mathrm{IID}_N$. We first polarize each pair (X^i, Y^i) to have statistical difference at most $1/3$ or mutual disjointness at least $\sqrt{2/3}$. Now, consider the pair (A, B) obtained by XORing (X_1, Y_1) and (X_2, Y_2) (in the sense of Lemma 2.10). Using this Lemma, we conclude that $(x_1, x_2) \in \mathrm{OR}(\Pi)_Y \Rightarrow (A, B) \in \mathrm{IID}_Y$ and that $(x_1, x_2) \in \mathrm{OR}(\Pi)_N \Rightarrow (A, B) \in \mathrm{IID}_N$. ■

6.5 Putting it Together

We can now prove that $\mathrm{SZK} \subseteq \mathrm{NISZK^h}$ and hence conclude the proof of Theorem 6.2. In the language of the previous section, Fact 6.3 says that the SZK-complete problem ED reduces to $\mathrm{AND}(\mathrm{OR}(\overline{\mathrm{EA}}, \mathrm{EA}))$ via a standard Karp (*i.e.*, many-one) reduction. Since EA and $\overline{\mathrm{EA}}$ are in $\mathrm{NISZK^h}$ (Lemma 6.4) and $\mathrm{NISZK^h}$ is closed under AND and OR (Lemma 6.10 and 6.11), we conclude that $\mathrm{ED} \in \mathrm{NISZK^h}$ and that $\mathrm{SZK} \subseteq \mathrm{NISZK^h}$.

An interesting corollary is the following new complete problem for SZK.

Corollary 6.12 IID *is complete for* SZK.

7 Computational Zero Knowledge

In this section, we extend the results presented in the previous section to computational zero knowledge. However, the techniques that we have used in the statistical case cannot be applied directly here, so we take a more indirect route to proving an equivalence for the computational case. We define the COMPUTATIONAL IMAGE INTERSECTION DENSITY CONDITION (CIIDC), a natural computational analogue of IID in the style of the INDISTINGUISHABILITY CONDITION and the CONDITIONAL PSEUDOENTROPY CONDITION used in [Vad] (see Section 3.2), and prove that all problems in ZK satisfy the CIIDC, building on our proof that every problem in SZK reduces to IID. Next we want to show that every problem in AM satisfying the CIIDC is in $\mathrm{NISZK^h}$. However, as the approach used in [BG] to show IID is in $\mathrm{NISZK^h}$ does not generalize to the computational case, following [Vad], we get around this difficulty by interpreting the COMPUTATIONAL IMAGE INTERSECTION DENSITY CONDITION as a special type of commitment scheme that is sufficient for constructing $\mathrm{NIZK^h}$ proofs. Hence, we show that any promise problem in $\mathrm{ZK} \cap \mathrm{AM}$ has a $\mathrm{NIZK^h}$ proof. For the other direction, we prove that ZK equals $\mathrm{ZK^h}$, a class which contains $\mathrm{NIZK^h}$, concluding that $\mathrm{NIZK^h} = \mathrm{ZK} \cap \mathrm{AM}$.

7.1 The Computational Image Intersection Density Condition

We define the Computational Image Intersection Density Condition, and show that any promise problem with a ZK proof satisfies this condition.

Definition 7.1 (Computational Image Intersection Density Condition (CIIDC)) *A promise problem Π satisfies* CIIDC *if there is a polynomial time mapping from strings $x \in \Pi$ to two distributions (X, Y) specified by circuits sampling from them such that*

1. *If $x \in \Pi_Y$, then X and Y are computationally indistinguishable.*
2. *If $x \in \Pi_N$, then (X, Y) have mutual disjointness at least $1/3$.*

Lemma 7.2 *Every promise problem $\Pi \in \mathrm{ZK}$ satisfies* CIIDC.

Proof: For $x \in \Gamma$, we know that x can be efficiently mapped via a reduction to IID to a pair (X_0, X_1) such that, on $x \in \Gamma_Y$, $\Delta(X_0, Y_0) < 2^{-n}$ and, on $x \in \Gamma_N$, $\mathrm{MutDisj}(X_0, X_1) > 1 - 2^{-n}$.

For $x \in \Theta$, we can apply [HILL] to the instance-dependent one-way function to obtain an instance-dependent pseudorandom generator $G_x(\cdot)$ with seed length $m = m(n)$ and arbitrary expansion $l = l(n)$. We consider $(G_x(U_m), U_l)$ and note that, on $x \in \Theta_Y$, $G_x(U_m)$ will be computationally indistinguishable from U_l, while, on $x \in \Theta_N$, the pair $(G_x(U_n), U_l)$ has disjointness $(1 - 2^{n-l})$. Applying Lemma 2.8, we obtain a pair of distributions Y_0, Y_1 such that, on $x \in \Theta_Y$, (Y_0, Y_1) are computationally indistinguishable and, on $x \in \Theta_N$, (Y_0, Y_1) have mutual disjointness $1/2 \cdot (1 - 2^{n-l})$ for large enough n.

Since it might not be possible to efficiently distinguish between instances in Γ and those in Θ, it is not sufficient to simply map x to (X_0, X_1) when $x \in \Gamma$, and to (Y_0, Y_1) when $x \in \Theta$. Rather, we map x to $(X, Y) = \mathrm{XOR}((X_0, X_1), (Y_0, Y_1))$.

For the YES instances, since on $x \in \Pi_Y = \Gamma_Y \cup \Theta_Y$ either (X_0, X_1) or (Y_0, Y_1) is computationally indistinguishable, the pair (X, Y) is computationally indistinguishable.[3] Additionally, on $x \in \Pi_N = \Gamma_N \cap \Theta_N$, by Lemma 2.9, (X, Y) will have mutual disjointness $1/2 \cdot (1 - 2^{-n}) \cdot (1 - 2^{n-l}) > 1/3$. Hence, Π satisfies the CIIDC. ■

7.2 Noninteractive, Instance-Dependent Commitments

We begin by reviewing Ben-Or and Gutfreund's [BG] proof that IID is in $\mathrm{NISZK^h}$ and note that this proof cannot be replicated in the computational case to show that every Π satisfying the CIIDC is in $\mathrm{NISZK^h}$. Ben-Or and Gutfreund show that IID is in $\mathrm{NISZK^h}$ by polarizing $(X_0, X_1) \in \mathrm{IID}$ to the distributions (Y_0, Y_1), setting the help string to $\sigma = Y_0(r)$ and having P prove to V that $\sigma \in \mathrm{Supp}(Y_1)$ by sending a random preimage in $Y_1^{-1}(\sigma)$. However, this protocol may fail to even

[3] Informally, assuming there is a distinguisher D for the XORed distributions, one could find a distinguisher D' for wlog, X_0 and X_1. In particular, D' can distinguish between $u \leftarrow X_0$ and $u \leftarrow X_1$ by choosing b at random, running D on u, Y_b and outputting b if $D(u) = 0$ and $\overline{b}$ otherwise. This strategy would maintain D's distinguishing advantage.

have completeness for promise problems satisfying CIIDC, since the images of Y_0 and Y_1 might even be disjoint, although they are computationally indistinguishable. Indeed, we do not expect to show that every problem satisfying CIIDC is in $\mathrm{NIZK^h}$, since $\mathrm{NIZK^h} \subseteq \mathrm{AM}$ but problems outside AM may satisfy CIIDC (indeed, if one-way functions exist, *every* promise problem satisfies the CIIDC). Thus, in showing an equivalence between interactive and noninteractive zero knowledge in the computational case, it is necessary to use a different approach. Following [Vad], we view IID/CIIDC as a kind of instance-dependent commitment scheme, and use it to implement the general construction of noninteractive zero-knowledge proofs for AM [FLS].

We show that promise problems that reduce to IID or that satisfy CIIDC have a natural form of noninteractive, instance-dependent commitment schemes. In particular, for a promise problem Π which reduces to IID (resp., satisfies the CIIDC), the sender and the receiver can use the Polarization Lemma to obtain a pair of distributions (Y_0, Y_1) that are statistically close on YES instances, and mutually disjoint on NO instances. To commit to a bit b, the sender draws c from Y_b and outputs c as the commitment. To reveal b, the sender only needs to prove that c is drawn from Y_b by presenting to the receiver the randomness used in sampling from Y_b.

Informally, the construction described above will satisfy the properties of a commitment scheme in an instance-dependent fashion. The scheme will be *hiding* on YES instances, since for $x \in \Pi_Y$, (Y_0, Y_1) are statistically close (resp., computationally indistinguishable), so it is hard to distinguish between commitments drawn from from Y_b and those drawn from $Y_{\overline{b}}$. Moreover, it is *binding* on NO instances, since for $x \in \Pi_N$, a negligible part of the images of Y_b intersect with the images of $Y_{\overline{b}}$, so there is a negligible probability an adversary can reveal both b and $\overline{b}$ as the bits it has committed to. Note that this binding property requires that the *sender generates the commitments honestly.* (Otherwise, it could always generate the commitment from the intersection of the supports, even if it negligibly small.) While assuming an honest sender is usually not suitable in applications of commitments, it turns out to be fine for constructing $\mathrm{NIZK^h}$ proofs, because the dealer generates the commitments.

We note that this commitment-based approach can also be used as an alternate, more circuitous proof of $\mathrm{NISZK^h} = \mathrm{SZK}$, since our results regarding commitments apply to both IID and CIIDC. Hence, the definitions and theorems presented below will deal with both the statistical and computational variants.

We now give a formal definition of the noninteractive, instance-dependent commitment schemes we will be using:

Definition 7.3 *A* noninteractive, instance-dependent commitment scheme *is a family* $\{\mathrm{Com}_x\}_{x\in\{0,1\}^*}$ *with the following properties:*

1. *The scheme* Com_x *proceeds in the stages: the* commit stage *and the* reveal stage. *In both stages, both the* sender *and the* receiver *share as common input the instance* x. *Hence we denote the sender and receiver as* S_x *and, respectively,* R_x*, and we write* $\mathrm{Com}_x = (S_x, R_x)$.

2. *At the beginning of the commit stage, the sender* S_x *receives as private input the bit* $b \in \{0, 1\}$ *to commit to. The sender then sends a single message* $c = S(x, b)$ *to the receiver.*

3. *In the reveal stage,* S_x *sends a pair* (b, d)*, where* d *is the* decommitment*string for bit* b. *Receiver* R_x *either accepts or rejects based on inputs* x, b, d *and* c.

4. *The sender S_x and receiver R_x algorithms are computable in time* $\text{poly}(|x|)$*, given the instance* x.

5. *For every $x \in \{0,1\}^*$, R_x will always accept (with probability* 1*) if both S_x and R_x follow their prescribed strategy.*

Security Properties. We now define the security properties of noninteractive, instance-dependent commitment schemes. These properties will be natural extensions of the hiding and binding requirements of standard commitments:

Definition 7.4 *A noninteractive, instance-dependent commitment scheme* $\text{Com}_x = (S_x, R_x)$ *is* statistically (resp., computationally) hiding on $I \subseteq \{0,1\}^*$ *if for every (resp., nonuniform PPT) R^*, the ensembles $\{S_x(0))\}_{x\in I}$ and $\{(S_x(1)\}_{x\in I}$ are statistically (resp., computationally) indistinguishable.*

For a promise problem $\Pi = (\Pi_Y, \Pi_N)$*, a noninteractive, instance-dependent commitment scheme* Com_x *is* statistically (resp., computationally) hiding on the YES instances *if* Com_x *is* statistically (resp., computationally) hiding on Π_Y.

Definition 7.5 *A noninteractive instance-dependent commitment scheme* $\text{Com}_x = (S_x, R_x)$ *is* statistically (resp., computationally) binding for honest senders on $I \subseteq \{0,1\}^*$ *if there exists a negligible function ε such that for all $x \in I$, a computationally unbounded (resp., nonuniform PPT) algorithm S^* succeeds in the following game with probability at most $\varepsilon(|x|)$:*

> *S outputs a commitment c. Then, given the coin tosses of S, S^* outputs pairs $(0, d_0)$ and $(1, d_1)$ and* succeeds *if in the reveal stage, $R_x(0, d_0, c) = R_x(1, d_1, c) =$* ACCEPT.

For a promise problem $\Pi = (\Pi_Y, \Pi_N)$*, a noninteractive, instance-dependent commitment scheme* Com_x *is* statistically (resp., computationally) binding for honest senders on the YES instances *if* Com_x *is* statistically (resp., computationally) binding on Π_Y.

Having defined noninteractive, instance-dependent commitment schemes, we proceed to show that they are equivalent to IID (resp., CIIDC), and consequently, SZK (resp., ZK).

Lemma 7.6 *A promise problem Π has a noninteractive, instance-dependent commitment scheme that is statistically (resp., computationally) hiding on* YES *instances and statistically binding for honest senders on* NO *instances if and only if Π reduces to* IID *(resp., if and only if Π satisfies the* CIIDC*).*

Proof: For the backwards direction, consider a problem Π that reduces to IID (the computational case will be similar). We construct the following protocol:

Commitment protocol for Π:

1. **Preprocessing:**

   First, reduce $x \in \Pi$ to an instance (X_0, X_1) of IID. Use the Polarization Lemma on (X_0, X_1) to obtain (Y_0, Y_1) such that, if $x \in \Pi_Y$, $\Delta(Y_0, Y_1) \le 2^{-n}$, and, if $x \in \Pi_N$, (Y_0, Y_1) have mutual disjointness $(1 - 2^{-n})$, where $n = |x|$.

2. **Commit Stage:**

   $S_x(x, b)$: To commit to bit $b \in \{0,1\}$, choose $d \xleftarrow{R} \{0,1\}^m$, where m is the input length of Y_b, set $c = Y_b(d)$ and output (c, d).

3. **Reveal Stage:**

   $R_x(x, c, b, d)$: Accept if and only if $Y_b(d) = c$.

On $x \in \Pi_Y$, we know that Y_0 and Y_1 have negligible statistical difference. Hence, a commitment to 1 is statistically indistinguishable from a commitment to 0. Hence, the scheme is computationally hiding on YES instances (actually, the scheme is statistically hiding.)

When $x \in \Pi_N$, the pair (Y_0, Y_1) has mutual disjointness $(1 - 2^{-n})$. It directly follows that only a negligible fraction of commitments can be opened in two ways.

In the case that we are working with a problem which satisfies the CIIDC, we use the same scheme. However, instead of polarizing, we will simply take direct products to amplify the mutual disjointness on NO instances while preserving computational indistinguishability on YES instances (Lemma 2.9).

For the forward direction, let $\mathrm{Com}_x = (S_x, R_x)$ be a noninteractive, instance-dependent commitment scheme that is statistically hiding on YES instances and statistically binding for honest senders on NO instances, and consider $X = S_x(0)$ and $Y = S_x(1)$:

- If $x \in \Pi_Y$, we know that $\Delta(\mathrm{view}_R(S_x(0), R), \mathrm{view}_R(S_x(1), R)) \leq \varepsilon(|x|)$, and hence, $\Delta(S_x(0), S_x(1)) \leq \varepsilon(|x|)$.
- If $x \in \Pi_N$, assume that there exists no negligible function $\mu(|x|)$ such that $\mathrm{MutDisj}(S_x(0), S_x(1)) = (1 - \mu(|x|))$. Hence for all negligible functions $\mu(|x|)$ and $c \leftarrow S_x(b)$, $\Pr\left[c \in S_x(\bar{b})\right] > \mu(|x|)$. But then, S can always succeed with probability greater than $\mu(|x|)$ at the game described in Definition 7.5. So, for some negligible μ, $(S_x(0), S_x(1))$ have mutual disjointness $(1 - \mu(|x|))$, and Π reduces to IID.

The proof for the computational case is analogous. ■

By combining our previous results concerning IID and CIIDC with Lemma 7.6, we obtain the following theorem:

Theorem 7.7 *If a promise problem Π is in SZK (resp.,* ZK*), then Π also has a noninteractive instance-dependent commitment scheme that is statistically (resp., computationally) hiding on* YES *instances and statistically binding for honest senders on* NO *instances.*

Proof: This follows from the fact that any $\Pi \in$ SZK (resp., ZK) reduces to IID (resp., satisfies CIIDC) (Lemma 7.2). By Lemma 7.6, Π has a noninteractive, instance-dependent commitment scheme. ■

7.3 From Noninteractive, Instance-Dependent Commitments to $\mathrm{NIZK}^{\mathrm{h}}$

In section, we will show that noninteractive, instance-dependent commitment schemes are sufficient to obtain $\mathrm{NIZK}^{\mathrm{h}}$. We start from the *hidden bits model*, a fictitious construction that implements noninteractive zero knowledge unconditionally for all promise problems in AM. Then, we show how our commitments can be employed in conjunction with this model to construct $\mathrm{NIZK}^{\mathrm{h}}$ proofs.

The Hidden Bits Model. The hidden bits model is a model due to Feige, Lapidot and Shamir [FLS] that allows for an unconditional construction of NIZK. It assumes that both the prover P and the verifier V share a common reference string σ, which we will call the hidden random string (HRS). However, only the prover can see the HRS. We can imagine that the individual bits of σ are locked in boxes, and only the prover has the keys to unlock them. The prover can selectively unlock boxes and reveal bits of the hidden random string. However, without the prover's help, the verifier has no information about any of the bits in the HRS.

Definition 7.8 (NIZK in the Hidden Bits Model [FLS]) *A* noninteractive zero knowledge proof system in the hidden-bits model *for a promise problem* Π *is a pair of probabilistic algorithms* (P, V) *(where* P *and* V *polynomial-time bounded) and a polynomial* $l(|x|) = |\sigma|$*, satisfying the following conditions:*

1. *Completeness. For all* $x \in \Pi_Y$*,* $\Pr[\exists(I, \pi) s.t.\ V(x, \sigma_I, I, \pi) = 1] \geq \frac{2}{3}$*, where* $(I, \pi) = P(x, \sigma)$*,* I *is a set of indices in* $\{0, \ldots, l(k)\}$*, and* σ_I *is the sequence of opened bits of* σ*,* $(\sigma_i : i \in I)$*, and where the probability is taken over* $\sigma \xleftarrow{R} \{0,1\}^{l(|x|)}$ *and the coin tosses of* P *and* V*.*
2. *Soundness. For all* $x \in \Pi_N$ *and all* P^**,* $\Pr[\exists(I, \pi) s.t.\ V(x, \sigma_I, I, \pi) = 1] \leq \frac{1}{3}$*, where* $(I, \pi) = P^*(x, \sigma)$*, where the probability is taken over* $\sigma \xleftarrow{R} \{0,1\}^{l(|x|)}$ *and the coin tosses of* P^* *and* V*.*
3. *Zero Knowledge. There exists a PPT* S *such that the ensembles of transcripts* $\{(x, \sigma, P(x, \sigma))\}_x$ *and* $\{S(x)\}_x$ *are statistically indistinguishable on* Π_Y*, where* $\sigma \xleftarrow{R} \{0,1\}^{l(|x|)}$*.*

Note that we have defined the zero-knowledge condition in this model to be statistical rather than computational. Indeed, the known construction of hidden bits NIZK proof systems is unconditional and yields statistically indistinguishable proof systems.

Theorem 7.9 ([FLS]) *Every promise problem* $\Pi \in \mathrm{NP}$ *has a hidden bits zero knowledge proof system* (P, V)*.*

As has been observed before (e.g. [Pas]), this construction for NP automatically implies one for all of AM.

Corollary 7.10 ([FLS]) *Every promise problem* $\Pi \in \mathrm{AM}$ *has a hidden bits zero knowledge proof system* (P, V)*.*

Proof: We will show this by transforming an AM proof into a statement that there exists some message from the prover that the verifier accepts. Since this statement is an NP statement, it can be proven in the hidden bits NIZK model.

Consider Π with an AM proof system (P', V'). We can assume that (P', V') have negligible completeness and soundness errors (this can be achieved by a polynomial number of parallel repetitions.) Let $p(|x|)$ be the length of the random challenge that V' sends to P', $q(|x|)$ be the length of V''s message. Consider the following promise problem Γ, which captures the completeness and soundness properties of (P', V'):

$$\begin{aligned} \Gamma_Y &= \{(x, r) : x \in \Gamma, r \in \{0,1\}^{p(|x|)}, \exists \text{ message } m \text{ such that } V'(x, r, m) = 1\} \\ \Gamma_N &= \{(x, r) : x \in \Gamma, r \in \{0,1\}^{p(|x|)}, \nexists \text{ message } m \text{ such that } V'(x, r, m) = 1\} \end{aligned}$$

It is clear that Γ is in NP, so there exists a hidden bits zero knowledge proof system (P'', V'') for it. Suppose the length of the hidden string is $l(|x|)$. Because of the vanishing completeness and soundness errors of (P', V'), we know that for a random choice of (x, r), with $x \in \Pi_Y$, the probability $(x, r) \in \Gamma_Y$ is exponentially close to 1. Similarly, if $x \in \Pi_N$ the probability $(x, r) \in \Gamma_N$ is exponentially close to 1.

We can build a hidden bits zero knowledge proof system (P, V) for Π in the following way. We let P and V share a hidden string σ of length $p(|x|) + l(|x|)$. P sets r to the first $p(|x|)$ bits of σ, and reveals them to V. Then, P uses the $l(|x|)$ remaining unrevealed hidden bits of σ to simulate P'''s hidden bits proof that $(x, r) \in \Gamma_Y$, and sends this simulated proof to V. V then simulates V'' and accepts if and only if V'' accepts.

Completeness and soundness follow from the completeness and soundness of (P', V') (as captured by Γ) and of (P'', V''). Finally, the zero knowledge of (P, V) is given by the zero knowledge of (P'', V''), and the fact that, for $x \in \Pi_Y$, $(x, r) \in \Gamma_Y$ with high probability ((P', V') has negligible completeness error). In particular, one can a simulator S for the proof system (P, V) by randomly selecting an r, and then using the simulator for (P'', V'') to produce proofs that $(x, r) \in \Gamma_Y$. ■

The corollary above shows that there exists an unconditional construction of NIZK for all problems in AM. However, this construction holds only in the impractical hidden bits model. In proving our results, we show how to implement this construction in the help model by exploiting a novel connection to noninteractive, instance-dependent commitment schemes:

Theorem 7.11 *If* $\Pi \in \mathrm{AM}$ *and* Π *has a noninteractive, honest-sender, instance-dependent commitment scheme that is statistically (resp., computationally) hiding on* YES *instances and statistically binding for honest senders on* NO *instances, then* $\Pi \in \mathrm{NISZK^h}$ *(resp.,* $\Pi \in \mathrm{NIZK^h}$*).*

Proof: Throughout the proof, we will assume that we have a computationally hiding commitment scheme, which we will use to build a $\mathrm{NIZK^h}$ proof system. The compiler used to build a $NISZKh$ proof system from statistically hiding commitments is identical. We show that we can use a noninteractive, honest-sender, instance-dependent commitment scheme to build a $\mathrm{NIZK^h}$ proof system which implements the hidden bits construction of [FLS]. Our general strategy will be to exploit the correspondence between the algorithms in our definition of an instance-dependent commitment scheme, and the three algorithms in a $\mathrm{NIZK^h}$ proof system. More specifically, we will have the dealer D use the sender algorithm to commit to a hidden bits string (this is why we can afford to assume the sender is honest). Since the prover P is allowed to be unbounded, we will use it to exhaustively search for openings to D's commitments. Finally, the verifier V will use the receiver algorithm to check P's openings.

Let $(P^{\mathrm{HB}}, V^{\mathrm{HB}})$ be a hidden bits proof system for Π and let $(\mathrm{Sen}, \mathrm{Rec})$ be the noninteractive, honest-sender bit commitment scheme for Π. Then, the following proof system (D, P, V) is $\mathrm{NIZK^h}$:

1. $D(x, 1^k)$: Select $\sigma^D \stackrel{R}{\leftarrow} \{0,1\}^m$, and run $\mathrm{Sen}(x, \sigma^D_i)$ to generate a commitment c_i, for all i. Output $c = (c_1, \ldots, c_m)$ as the public help parameter.

2. $P(x, c)$: Exhaustively find a random opening o^P_i for each c_i (and, implicitly, each σ^D_i). If one commitment c_i can be opened as both 0 or 1, P outputs o^P_i according to the distribution $O|_{C=c_i}$, where (O, C) is the output of S on a random bit b. Let σ^P be the secret string obtained by P opening D's help string. P runs $P^{\mathrm{HB}}(x, \sigma^P)$ to obtain (I, π). Send $(I, \sigma^P_I, o^P_I, \pi)$ to V.

3. $V(x, I, o_I^P, \pi)$: Compute $\sigma_j^P, \forall j \in I$. Use Rec to check that the commitments are consistent. Run $V^{\mathrm{HB}}(x, I, \sigma_I^P, \pi)$ and accept if and only if V^{HB} accepts.

The reason our protocol refers to 2 secret strings (σ^D and σ^P) is that our commitments are not necessarily binding on YES instances. Consequently, P might not be able to uniquely recover the same secret string σ^D based on D's help string consisting of the commitments to σ^D. That is why we have P recreate another secret string σ^P by drawing from the distribution of bits conditioned on the help string. We note that:

- This only happens for YES instances. For NO instances, P has a negligible chance of being able to open a σ^P different from σ^D. This guarantees that the potential ambiguity of the help string cannot affect soundness.
- The distributions (σ^D, c) and (σ^P, c) are identically distributed (the only difference is the order in which σ and c are drawn).

We now show the protocol described above satisfies the conditions necessary for it to be a $\mathrm{NIZK}^{\mathrm{h}}$ proof system:

1. *Completeness.* This follows from the completeness of the hidden bits system $(P^{\mathrm{HB}}, V^{\mathrm{HB}})$.
2. *Soundness.* We show that a potentially malicious prover P^* can open σ^D in only one way with overwhelming probability. Since the commitment scheme is statistically binding on NO instances, the probability that a commitment c_i can be opened as both 0 and 1 will be some negligible function $\varepsilon(n)$, where $n = |x|$. Hence, the probability that any commitment c_i can be opened in two ways is at most $m \cdot \varepsilon(n)$. Assuming that there existed a cheating P^* that could convince V to accept with probability p, then we can obtain a cheating $(P^*)^{\mathrm{HB}}$ which outputs accepting proofs with probability at least $p - m\varepsilon(n)$, by defining $(P^*)^{\mathrm{HB}}(x, \sigma) = P^*(x, c)$ where $(c_1, \ldots, c_m) = (\mathrm{Sen}(x, \sigma_1), \ldots, \mathrm{Sen}(x, \sigma_m))$. Since $(P^*)^{\mathrm{HB}}$ can produce an accepting transcript with only negligible probability, P^* produces an accepting proof with negligible probability. Therefore, the soundness of $(P^{\mathrm{HB}}, V^{\mathrm{HB}})$ carries over to (D, P, V).
3. *Zero Knowledge.* We construct the following simulator S for the proof system. We let S be a pair of PPTs (S^{HB}, S'), where S^{HB} is the simulator for the hidden bits NIZK proof system for Π. S^{HB} takes in as input $x \in \Pi$, and outputs (σ_I, I, π). S' takes in σ_I as input, randomly completes σ by selecting the bits not in σ_I, and generates commitment/opening pairs (c_i, o_i) for all bits σ_i (the pairs are drawn randomly from the possible choices of commitments and openings).

   In order to show that S can truly simulate real transcripts, we first build the following distributions:

   - The distributions of real transcripts, generated by the dealer D and the prover P:
     $H_0 = \{c \leftarrow D(1^k), (I, \sigma^P, o^P, \pi) \leftarrow P(x, c) : (c, \sigma_I^P, I, o_I^P, \pi)\}$
   - A hybrid for which a modified dealer D' not only sends c, but also the openings o to the prover P^{HB}.
     $H_1 = \{(\sigma^D, c, o^D) \leftarrow D'(1^k), (I, \pi) \leftarrow P^{\mathrm{HB}}(x, \sigma^D) : (c, \sigma_I^D, I, o_I^D, \pi)\}$

- A hybrid where σ is generated uniformly and fed to P^{HB} to produce (I, π), as well as to a modified dealer D'', which on input σ, x produces the pair (c, o) for σ.
$H_2 = \{\sigma \leftarrow \{0,1\}^m, (I, \pi) \leftarrow P^{\mathrm{HB}}(x, \sigma), (c, o) \leftarrow D''(\sigma, x) : (c, \sigma_I, I, o_I, \pi)\}$
- The distribution of simulated transcripts:
$H_3 = \{(\sigma_I, I, \pi) \leftarrow S^{\mathrm{HB}}, (\sigma \backslash \sigma_I, c, o) \leftarrow S'(\sigma_I, I) : (c, \sigma_I, I, o_I, \pi)\}$
where by $\sigma \backslash \sigma_I$ we refer to those bits of σ which had not already been selected by the choice of σ_I.

We now proceed to prove the indistinguishability relationships between these different hybrids. By examination, we see that H_0, H_1 and H_2 are identically distributed. By the properties of hidden bits zero knowledge proof systems, we know that the transcripts produced by P^{HB}, $\{\sigma \leftarrow \{0,1\}^m, (I, \pi) \leftarrow P^{\mathrm{HB}}(x, c) : (\sigma_I, I, \pi)\}$ are statistically indistinguishable from those simulated by S^{HB}, $\{(\sigma_I, I, \pi) \leftarrow S^{\mathrm{HB}} : (\sigma_I, I, \pi)\}$, so the σ_I, I, π fragments of the hybrids H_1 and H_2 are statistically indistinguishable. In both cases, the commitments c_I and openings o_I to the bits in σ_I are generated using the sender algorithm Sen, so the distributions remain statistically indistinguishable if we include these. The distributions differ, however, in how the other commitments $c \backslash c_I$ are generated. In H_2, these are commitments to bits $\sigma \backslash \sigma_I$ that are correlated with (σ_I, I, π). In H_3, they are commitments to bits $\sigma \backslash \sigma_I$ that are uniform and independent of (σ_I, I, π). But, by the hiding property, commitments to any two sequences of bits are computationally indistinguishable. Hence H_0 and H_3, representing the real and, respectively, the simulated transcripts, are computationally indistinguishable, proving that the proof system (D, P, V) is zero knowledge.

If the commitment scheme is statistically rather than computationally hiding on NO instances, then the ensembles above are statistically indistinguishable, and we obtain a $\mathrm{NISZK}^{\mathrm{h}}$ proof system.

■

Remarks. We make the following observations about the protocol in the proof of Theorem 7.11.

1. If the commitment scheme is not instance-dependent, but rather depends only on the security parameter (i.e., the length of the input x), then we obtain a proof system in the *public parameter* model. Combining this with the construction of commitments from one-way functions [HILL, Nao], we get another proof of the fact that one-way functions imply $\mathrm{NIZK}^{\mathrm{pub}} = \mathrm{AM}$ [BG, Pas]. We note that Pass and shelat [Pas] actually achieve the stronger property of *adaptive* zero knowledge.

2. The protocol requires a computationally unbounded honest prover, because the prover must break the commitments. However, the prover can be implemented efficiently in a generalization of the help model where the dealer can generate secret information (e.g. the openings to the commitments) for the prover in addition to the common reference string. Such a model can be useful for applications of noninteractive zero knowledge where the dealer and the honest prover are the same party, such as the Bellare–Goldwasser signature scheme [BG]. (This signature scheme also requires that the zero knowledge property holds even when *many*, *adaptively chosen* statements are proven using the same reference string; unfortunately, our construction does not provide such guarantees.) This model for noninteractive zero knowledge

should be contrasted with one where the *verifier* receives secret information from the dealer, which has proven useful in the construction of encryption schemes secure against chosen-ciphertext attack [CS], and one where both parties receive secret information, as studied in [CD].

7.4 From $\mathrm{ZK^h}$ to ZK

In this section, we generalize the results of Ben-Or and Gutfreund [BG] that $\mathrm{SZK^h} = \mathrm{SZK}$ (Theorem 3.3) to show that adding help to ZK proofs does not confer any additional power:

Theorem 7.12 (Theorem 1.3, restated) $\mathrm{ZK^h} = \mathrm{ZK}$.

To prove Theorem 3.3, Ben-Or and Gutfreund employ the techniques of [AH, PT, GV], by considering the output of the simulator S for a zero-knowledge proof for Π as the moves of a *virtual prover* and a *virtual verifier*. The simulated transcripts are compared to the transcripts output by a cheating strategy for a real prover P_S (called the *simulation-based prover*), which tries to imitate the behavior of the virtual prover. Intuitively, on YES instances, the output of the simulator should be statistically close to the output of the simulation-based prover interacting with the real verifier. On NO instances, however, if we modify the simulator to accept with high probability (we can easily modify it to do that), the difference between the two transcripts must be significant. [BG] exploit this to show that any problem in $\mathrm{SZK^h}$ can be reduced to the intersection of the SZK-complete problems Statistical Difference([SV2]) and Entropy Difference([GV]). Since the other direction ($\mathrm{SZK} \subseteq \mathrm{SZK^h}$) follows from the definitions, the conclusion that $\mathrm{SZK} = \mathrm{SZK^h}$ follows immediately. We will use the same strategy with $\mathrm{ZK^h}$, replacing statistical measures of closeness with computational ones. To do this, we replace the SZK-complete problems SD and ED with the Indistinguishability Condition and the Conditional Pseudoentropy Condition, which characterize the class ZK, and show that for every $\Pi \in \mathrm{ZK^h}$, Π can be reduced to the intersection of a problem which satisfies Indistinguishability Condition and a problem which satisfies Conditional Pseudoentropy Condition, and is thus in ZK.

We will use the following notation throughout this section: we let (D, P, V) be a ZK proof system for promise problem Π, and we let S be the simulator for the honest verifier V. We assume that the verifier uses a total of $r = r(|x|)$ coins. Including the dealer's message, we assume that $2l$ messages make up a transcript, where $l = l(|x|)$, and that each message has length r. Additionally, the last message reveals the verifier's random coins. We use the notation $S(x)$ to refer to the simulated transcripts. For a transcript γ, we denote γ_i the prefix of γ consisting of the first i messages.

We construct the simulation-based prover in the following manner: for an odd i, given a conversation prefix $\gamma \in \{0,1\}^{(i-1)r}$, the next message of P_S is:

1. If the probability that $S(x)$ outputs a conversation with prefix γ is 0, then P_S sends a dummy message, say 0^r.

2. Otherwise, P_S replies with the same conditional probability as the virtual prover, sending β with probability $\Pr[S(x)_i = \gamma\beta | S(x)_{i-1} = \gamma]$.

Note that P_S sends the first message instead of the dealer, using the simulator to generate the help string. Define $\langle P_S, V\rangle(x)$ to be the distribution of the possible transcripts of conversations between P_S and V.

Lemma 7.13 ([AH, PT, GV, BG]) *For all x,* $\mathrm{KL}(S(x)|\langle P_S, V\rangle(x)) = r - \sum_{i=1}^{l}[\mathrm{H}(S(x)_{2i}) - \mathrm{H}(S(x)_{2i-1})]$.

Lemma 7.14 ([AH, PT, GV, BG]) *For $x \in \Pi_N$, let p denote the probability that $S(x)$ outputs an accepting transcript. Suppose that $\Delta(D(x), S(x)_1) \leq q_1$. Denote by $q_2 = q_2(|x|)$ the soundness of the protocol. Let $q = 2q_1 + q_2$, and suppose that $p \geq q$. Then,*

$$\mathrm{KL}(S(x)|\langle P_S, V\rangle(x)) \geq \mathrm{KL}_2(p, q).$$

We will use the previous two lemmas to prove the main result of this section:

Proof of Theorem 7.12 Since $\mathrm{ZK} \subseteq \mathrm{ZK}^{\mathrm{h}}$ by definition, we prove $\mathrm{ZK}^{\mathrm{h}} \subseteq \mathrm{ZK}$. Consider a problem Π with a ZK^{h} proof system with completeness and soundness errors at most $(2lr)^{-2}/2$. We modify the proof system such that 0^{2lr} is always an accepting transcript, and such that the simulator always outputs accepting transcripts (e.g., swap on rejecting transcripts with 0^{2lr}). The new proof system has soundness error at most $2^{-r} + (2lr)^{-2}/2$.

Similarly to [BG, GV, Vad], consider the following distributions:

- $X_{x,1} = (S(x)_2, \ldots S(x)_{2l})$, $Y_{1,x} = (S(x)_1, \ldots S(x)_{2l-1})$.
- $X_{x,2} = D(x)$, $Y_{2,x} = S(x)_1$.

Claim 7.15 *If $x \in \Pi$, $X_{2,x} \stackrel{c}{\equiv} Y_{2,x}$ and $(X_{1,x}, Y_{1,x}) \stackrel{c}{\equiv} (X', Y')$, where $\mathrm{H}(X'|Y') = r$.*

Proof: When $x \in \Pi_Y$, $X_{2,x} \stackrel{c}{\equiv} Y_{2,x}$ and $(X_{1,x}, Y_{1,x}) \stackrel{c}{\equiv} (X', Y')$, where (X', Y') is the distribution of real transcripts produced by $\langle D, P, V\rangle$. That is, $X' = (\langle D, P, V\rangle(x)_2, \ldots \langle D, P, V\rangle(x)_{2l})$ and $Y' = (\langle D, P, V\rangle(x)_1, \ldots \langle D, P, V\rangle(x)_{2l-1})$.

The conditional entropy of X' given Y' will be:

$$\mathrm{H}(X'|Y') = \sum_{i=1}^{l} \mathrm{H}(\langle D, P, V\rangle(x)_{2i}|\langle D, P, V\rangle(x)_{2i-1}) = r$$

since the sum measures the total entropy contributed by the verifier's messages. ■

Claim 7.16 *If $x \in \Pi$, either $\Delta(X_{2,x}, Y_{2,x}) \geq (2lr)^{-1}$ or $\mathrm{H}(X_{1,x}|Y_{1,x}) \leq r - 1$.*

Proof: Assume $\Delta(X_{2,x}, Y_{2,x}) \leq (2lr)^{-1}$. Then, we have:

$$\begin{aligned}
&\mathrm{H}(X_{1,x}|Y_{1,x}) \\
&= \sum_{i=1}^{l} \mathrm{H}(S(x)_{2i}|S(x)_{2i-1}) \\
&= \sum_{i=1}^{l} \mathrm{H}(S(x)_{2i}) - \mathrm{H}(S(x)_{2i-1})) \\
&= r - \mathrm{KL}(S(x)|\langle P_S, V\rangle(x)) \text{ (by Lemma 7.13)} \\
&\leq r - \mathrm{KL}_2(1, 1/2) \text{ (by Lemma 7.14, with } p = 1, q_1 = (2lr)^{-1}, q_2 = 2^{-l} + (2lr)^{-2}/2, q = q_1 + 2q_2 \leq 1/2) \\
&= r - \log 2 \\
&= r - 1
\end{aligned}$$

■

Having mapped instances $x \in \Pi$ to $(X_{x,1}, Y_{x,1})$ and $(X_{x,2}, Y_{x,2})$, consider the promise problems Γ and Λ defined by $\Gamma_Y = \Lambda_Y = \Pi_Y, \Gamma_N = \{x \in \Pi_N : \Delta(X_{x,2}, Y_{x,2}) \geq (2lr)^{-1}\}$ and $\Lambda_N = \{x \in \Pi_N : \mathrm{H}(X_{x,1}, Y_{x,1}) \leq r - 1\}$. Then $\Pi = \Gamma \cap \Lambda$ (i.e., $\Pi_Y = \Gamma_Y \cap \Lambda_Y$ and $\Pi_N = \Gamma_N \cup \Lambda_N$). Since ZK is closed under intersection (run protocols for Γ and Λ in parallel), it suffices to show that both $\Gamma \in$ ZK and $\Lambda \in$ ZK. Both Γ and Λ are in IP; this follows because they are restrictions of Π, which is in $\mathrm{ZK}^{\mathrm{h}} \subseteq \mathrm{IP}$. Γ satisfies the Indistinguishability Condition (the inverse polynomial statistical difference can be amplified to 2/3 by taking direct products), so $\Gamma \in$ ZK (by Theorem 3.6), and Λ satisfies the Conditional Pseudoentropy Condition, so $\Lambda \in$ ZK (by Theorem 3.8). Consequently $\Pi \in \mathrm{ZK} \subseteq \mathrm{IP}$.

7.5 Putting It Together

We can now use the previous sections' results to prove our main theorems regarding computational zero knowledge:

Theorem 7.17 (Theorem 1.1, restated) $\mathrm{ZK}^{\mathrm{h}} \cap \mathrm{AM} = \mathrm{ZK} \cap \mathrm{AM} = \mathrm{NIZK}^{\mathrm{h}}$.

Proof: By definition, $\mathrm{NIZK}^{\mathrm{h}} \subseteq \mathrm{ZK}^{\mathrm{h}} \cap \mathrm{AM}$. For the other direction, we know any $\Pi \in$ ZK has a noninteractive, instance-dependent commitment scheme (Theorem 7.7), so a $\mathrm{NIZK}^{\mathrm{h}}$ proof can built for Π (Theorem 7.11). Hence, $\mathrm{ZK}^{\mathrm{h}} \cap \mathrm{AM} \subseteq \mathrm{NIZK}^{\mathrm{h}}$, which completes the proof of our theorem. ■

Theorem 7.18 $\Pi \in \mathrm{ZK} = \mathrm{ZK}^{\mathrm{h}}$ *if and only if* $\Pi \in$ IP *and* Π *satisfies the* CIIDC.

Proof: Since a promise problem that satisfies the CIIDC also satisfies the Indistinguishability Condition (this follows from the fact that of two distributions have disjointness α, they must have statistical difference at least α), the promise problem must have a ZK proof system by Theorem 3.6. Conversely, any problem in $\mathrm{ZK}^{\mathrm{h}} = \mathrm{ZK}$ satisfies CIIDC by Lemma 7.2. ■

8 Quantum Statistical Zero Knowledge

In this section, we study different variants of help for quantum noninteractive statistical zero knowledge. We start by providing complete problems for the class QNISZK defined by Kobayashi [Kob] and proceed to define the following two types of help: *pure quantum help* and *mixed quantum help*.

8.1 Complete problems for QNISZK

Kobayashi [Kob] gave a complete problem for the class of quantum noninteractive perfect zero-knowledge, but not for statistical zero-knowledge. We continue this line of work and give two complete problems for QNISZK, Quantum Entropy Approximation (QEA) and Quantum Statistical Closeness to Uniform (QSCU).

Let ρ be a quantum mixed state of n qubits which can be created in time polynomial in n by a quantum machine and t a positive integer. Then,

$$\mathrm{QEA}_Y = \{(\rho, t) : S(\rho) \geq t + 1\} \qquad \mathrm{QSCU}_Y = \{\rho : ||\rho - \mathcal{U}|| \leq 1/n\}$$
$$\mathrm{QEA}_N = \{(\rho, t) : S(\rho) \leq t - 1\} \qquad \mathrm{QSCU}_N = \{\rho : ||\rho - \mathcal{U}|| \geq 1 - 1/n\}$$

Note that these problems are the quantum equivalents of EA and SCU where the statistical difference is replaced by the trace distance and the Shannon entropy by the von Neumann entropy.

Ben-Aroya and Ta-Shma showed that QEA reduces to Quantum Statistical Difference QSD. In fact, during their proof, they showed that QEA reduces to $\mathrm{QSCU}^{a,b}$ for some parameters a, b but these parameters a, b are not good enough to show that QEA $\in$ QNISZK. We extend their proof to show that QEA $\in$ QNISZK and then conclude using similar techniques than the ones used in the classical case (see [GSV2] as well as the analysis of QNISZK done by Kobayashi [Kob]). The proof follows from the following lemmas.

Lemma 8.1 QEA $\in$ QNISZK.

Proof: Let (X, t) an instance of QEA with m input qubits and $\mathcal{U}$ the totally mixed distribution.

Claim 8.2 ([BT]) *From (X, t) We can create X' in quantum polynomial time such that*

- $(X, t) \in \mathrm{QEA}_Y \Rightarrow \Delta(X', \mathcal{U}) \leq 5\varepsilon$
- $(X, t) \in \mathrm{QEA}_N \Rightarrow \Delta(X', \mathcal{U}) \geq \frac{1}{2qm}$

for any q such that $q \geq 2\log(1/\varepsilon) + \log(qm) + O(1)$ and also $q \geq \sqrt{\log(1/\varepsilon)}\sqrt{q}n + 1$.

We apply this claim with the following parameters : fix $\varepsilon = 2^{-k}$ with $k \in \mathrm{poly}(n)$ and then $q \in \mathrm{poly}(n)$ that satisfies the constraints. Let X' be the resulting distribution. Now let $r = 8k \cdot (qm)^2 \in \mathrm{poly}(n)$ and $Y = X'^{\otimes r}$. By using bounds on Statistical Difference, we have

- $X \in \mathrm{QEA}_Y \Rightarrow \Delta(X', \mathcal{U}) \leq 5r\varepsilon \leq 2^{-\Omega(k)}$
- $X \in \mathrm{QEA}_N \Rightarrow \Delta(X', \mathcal{U}) \geq 1 - 2^{-k}$

Thus, QEA reduces to $\mathrm{QSCU}^{\mu(n),1-\mu(n)}$ for some negligible function μ. Kobayashi showed in [Kob] that $\mathrm{QSCU}^{\mu(n),1-\mu(n)} \in$ QNISZK for every negligible function μ thus we conclude that QEA $\in$ QNISZK. ■

Lemma 8.3 QSCU *reduces to* QEA.

Proof: We use the following fact about the relation of trace distance and von Neumann entropy

Fact 8.4 *Let X be a quantum state of dimension n.*

1. $\|X - \mathcal{U}\|_{tr} \leq \alpha \Rightarrow S(X) \geq n \cdot (1 - \alpha - 1/2^n)$.
2. $\|X - \mathcal{U}\|_{tr} \geq \beta \Rightarrow S(X) \leq n - \log(\frac{1}{1-\beta})$.

Let X a quantum mixed state of dimension n.

If $n \geq 16$. $\|X - \mathcal{U}\|_{tr} \leq 1/n \Rightarrow S(X) \geq n - 2$. $\|X - \mathcal{U}\|_{tr} \geq 1 - 1/n \Rightarrow S(X) \leq n - 4$. In this case, the reduction from QSCU to QEA since $X \in \mathrm{QSCU}_Y \Rightarrow X \in \mathrm{QEA}_Y$ and similarly, $X \in \mathrm{QSCU}_N \Rightarrow X \in \mathrm{QEA}_N$.

When $n < 16$, we can determine whether $X \in \mathrm{QSCU}_Y$ or $X \in \mathrm{QSCU}_N$ in polynomial time. We can therefore easily create in quantum polynomial time a distribution X' such that $X \in \mathrm{QSCU}_Y \Rightarrow X' \in \mathrm{QEA}_Y$ and $X \in \mathrm{QSCU}_N \Rightarrow X' \in \mathrm{QEA}_N$.

From this construction, we conclude that QSCU reduces to QEA. ■

It is easy to prove that QSCU is hard for QNISZK by naturally extending the results of Kobayashi [Kob]. It follows that

Theorem 8.5 QEA *and* QSCU *are complete for* QNISZK.

Proof: QSCU is hard for QNISZK and QSCU reduces to QEA so both problems are hard for QNISZK. QEA $\in$ QNISZK and QSCU $\preccurlyeq$ QEA so they are both in QNISZK. ■

8.2 Help in Quantum Noninteractive Zero-Knowledge

In quantum noninteractive zero knowledge, the only model we defined so far is the model where the prover and the verifier share the maximally entangled state $\sum_i |i\rangle_P |i\rangle_V$ which can be created by a dealer with quantum polynomial power ([Kob]). In the previous section, we provided two complete problems for this class. Here, we extend this definition to allow the dealer to create as help a quantum state that depends on the input.

We define two types of help and study the resulting classes:

- *Pure Help*: In the usual framework of quantum zero-knowledge protocols, the prover and the verifier use only unitaries. We define $\mathrm{QNISZK}^{\mathrm{h}}$ as the class where the prover and the verifier share a pure state (*i.e.,* the outcome of a unitary operation) created by the dealer in quantum polynomial time. This state can depend on the input. Note that since the maximally entangled state is a pure state $\mathrm{QNISZK} \subseteq \mathrm{QNISZK}^{\mathrm{h}}$. In fact, we show that $\mathrm{QNISZK}^{\mathrm{h}} = \mathrm{QSZK} = \mathrm{QSZK}^{\mathrm{h}}$.
- *Mixed Help*: The previous definition does not allow the dealer to have some private coins and hence does not fully correspond to $\mathrm{NISZK}^{\mathrm{h}}$. We suppose now that the prover and verifier share a *mixed* quantum state created by the dealer. As before, the dealer has quantum polynomial power and the state depends on the input. We call the resulting class $\mathrm{QNISZK}^{\mathrm{mh}}$ and show that this kind of help is most probably stronger than quantum interaction.

For these classes, the definition of the zero knowledge property remains the same as in the case of QNISZK (Section 5).

8.2.1 Pure Help.

We suppose here that there is a trusted dealer with quantum polynomial power. On input x, he performs a unitary D_x and creates a pure state $D_x(|0\rangle) = |h_{PV}\rangle$ in the space $\mathcal{P} \times \mathcal{V}$. The prover gets $h_P = \mathrm{Tr}_{\mathcal{V}}(h_{PV})$ and the verifier gets $h_V = \mathrm{Tr}_{\mathcal{P}}(h_{PV})$. Note that the state h_{PV} is a pure state and depends on the input.

Definition 8.6 *We say that* $\Pi \in \mathrm{QSZK}^{\mathrm{h}}$ *(resp.* $\Pi \in \mathrm{QNISZK}^{\mathrm{h}}$*) if there is an interactive (resp. noninteractive) protocol* $\langle D, P, V\rangle$ *that solves* Π*, has the zero knowledge property and where the verifier and the prover share a pure state* h_{PV} *created by a dealer* D *that has quantum polynomial power and access to the input. They also start with an arbitrary polynomial number of qubits initialized at* $|0\rangle$*.*

Next, we prove a quantum analogue of Theorem 6.2, *i.e.,* interactive and noninteractive zero knowledge are equivalent in the pure help model. We remark that the proof of this statement is much more straightforward than in the classical case.

Theorem 8.7 $\mathrm{QNISZK^h} = \mathrm{QSZK} = \mathrm{QSZK^h}$

Proof: We start by showing that $\mathrm{QSZK^h} \subseteq \mathrm{QSZK}$ (and hence by definition $\mathrm{QNISZK^h} \subseteq \mathrm{QSZK}$). Let $\Pi \in \mathrm{QSZK^h}$ and $\langle D, P, V\rangle$ denote the protocol. Since h_{PV} is a pure state, we can create another protocol $\langle \widetilde{P}, \widetilde{V}\rangle$ where the verifier takes the place of the dealer. That is, V generates for his first message the state $|h_{PV}\rangle$ and sends the h_P part to the dealer while keeping the h_V part for himself. At this point, note that the verifier and prover have exactly the same states then when the dealer generates the state $|h_{PV}\rangle$ and sends it to them.

The protocol is the same so soundness and completeness are preserved. The first message in $\langle \widetilde{P}, \widetilde{V}\rangle$ can be simulated because the circuit of the dealer is public and computable in quantum polynomial time. The remaining messages in $\langle \widetilde{P}, \widetilde{V}\rangle$ can be simulated because of the zero-knowledge property of the protocol $\langle D, P, V\rangle$.

The inclusion $\mathrm{QSZK} \subseteq \mathrm{QNISZK^h}$ (and hence by definition $\mathrm{QSZK} \subseteq \mathrm{QSZK^h}$) follows immediately from Watrous' two-message protocol for the QSZK-complete problem QSD [Wat1]. The first message of the verifier can be replaced by the dealer's help. ∎

8.2.2 Mixed help.

In the most general case, the dealer can create as help a mixed quantum state, *i.e.*, a state that can depend on some private coins or measurements as well as the input.

Definition 8.8 *We say that* $\Pi \in \mathrm{QNISZK^{mh}}$ *if there is a noninteractive protocol* $\langle D, P, V\rangle$ *that solves* Π *with the zero-knowledge property, where the verifier and the prover share a mixed state* h_{PV} *created by a dealer* D *that has quantum polynomial power and access to the input. They also start with* $|0\rangle$ *qubits.*

Note that the only difference between $\mathrm{QNISZK^h}$ and $\mathrm{QNISZK^{mh}}$ is that the verifier and the prover share a mixed state instead of a pure state; however, we show that this difference is significant. In the classical case, a model was studied where the dealer flips some coins r and sends correlated messages $m_P(r)$ and $m_V(r)$ to the prover and the verifier. The resulting class was called $\mathrm{NISZK^{sec}}$ and it was shown by Pass and shelat in [Pas] that $\mathrm{NISZK^{sec}} = \mathrm{AM}$. To create the secret correlated messages $m_P(r)$ and $m_V(r)$ in our quantum setting, we just have to create the following state : $|\phi\rangle = \sum_r |r\rangle|m_P(r)\rangle|m_V(r)\rangle$. This state can be created in polynomial time because $m_P(r)$ and $m_V(r)$ can be created with a classical circuit. The dealer keeps the r part, sends the m_P part to the prover and the m_V part to the verifier. From this construction, we can easily see that $\mathrm{AM} = \mathrm{NISZK^{sec}} \subseteq \mathrm{QNISZK^{mh}}$. Note that it is not known that $\mathrm{NP} \subseteq \mathrm{QSZK} = \mathrm{QNISZK^h}$ so this may be interpreted as evidence that $\mathrm{QNISZK^h}$ is a strict subset of $\mathrm{QNISZK^{mh}}$.

Last, when we also allow the verifier to use non-unitary operations (*i.e.*, private coins and measurements), we don't know if help and interaction are equivalent. The case of quantum zero knowledge protocols with non-unitary players is indeed very interesting and we refer the reader to [CK1] for more results.

Acknowledgements. We thank the anonymous referees for their helpful comments.

References

[AH] W. Aiello and J. Håstad. Statistical Zero-Knowledge Languages Can Be Recognized in Two Rounds. *Journal of Computer and System Sciences*, 42(3):327–345, June 1991.

[BM] L. Babai and S. Moran. Arthur-Merlin Games: A Randomized Proof System and a Hierarchy of Complexity Classes. *Journal of Computer and System Sciences*, 36:254–276, 1988.

[BG] M. Bellare and S. Goldwasser. New Paradigms for Digital Signatures and Message Authentication Based on Non-Interactive Zero Knowledge Proofs. In *CRYPTO '89*, pages 194–211, 1989.

[BMO] M. Bellare, S. Micali, and R. Ostrovsky. Perfect zero-knowledge in constant rounds. In *STOC '90: Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 482–493, 1990.

[BT] A. Ben-Aroya and A. Ta-Shma. Quantum expanders and the quantum entropy difference problem. *ArXiv Quantum Physics e-prints, quant-ph/0702129*, 2007.

[BGG+] M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway. Everything Provable is Provable in Zero-Knowledge. In *CRYPTO '88*, pages 37–56, 1988.

[BG] M. Ben-Or and D. Gutfreund. Trading Help for Interaction in Statistical Zero-Knowledge Proofs. *Journal of Cryptology*, 16(2), March 2003. Preliminary version appeared as [GB].

[BDMP] M. Blum, A. De Santis, S. Micali, and G. Persiano. Noninteractive Zero-Knowledge. *SIAM Journal on Computing*, 20(6):1084–1118, Dec. 1991.

[BFM] M. Blum, P. Feldman, and S. Micali. Non-Interactive Zero-Knowledge and Its Applications (Extended Abstract). In *STOC '88: Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 103–112, 1988.

[BCC] G. Brassard, D. Chaum, and C. Crépeau. Minimum Disclosure Proofs of Knowledge. *Journal of Computer and System Sciences*, 37(2):156–189, Oct. 1988.

[CK1] A. Chailloux and I. Kerenidis. Increasing the power of the verifier in Quantum Zero Knowledge. *Arxiv Quantum Physics e-prints, quant-ph/07114032*, 2007.

[CK2] A. Chailloux and I. Kerenidis. The role of help in Classical and Quantum Zero-Knowledge. Cryptology ePrint Archive, Report 2007/421, 2007. `http://eprint.iacr.org/`.

[Cio] D. Ciocan. Constructions and Characterizations of Non-Interactive Zero-Knowledge. Undergradute thesis, Harvard University, 2007.

[CV] D. F. Ciocan and S. Vadhan. Interactive and Noninteractive Zero Knowledge Coincide in the Help Model. Cryptology ePrint Archive, Report 2007/389, 2007. `http://eprint.iacr.org/`.

[CD] R. Cramer and I. Damgaard. Secret-Key Zero-Knowledge and Non-Interactive Verifiable Exponentiation. In *ACR Theory of Cryptography Conference (TCC '04)*, pages 223–237. Springer-Verlag, 2004.

[CS] R. Cramer and V. Shoup. Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption. In *EUROCRYPT '02: Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, pages 45–64, London, UK, 2002. Springer-Verlag.

[DDPY] A. De Santis, G. De Crescenzo, G. Persiano, and M. Yung. On Monotone Formula Closure of SZK. In *Proc. 26th ACM Symp. on Theory of Computing*, pages 454–465, Montreal, Canada, 1994. ACM.

[FLS] U. Feige, D. Lapidot, and A. Shamir. Multiple Non-Interactive Zero Knowledge Proofs Under General Assumptions. *SIAM Journal on Computing*, 29(1):1–28, 1999.

[GMW] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity, or All languages in NP have zero-knowledge proof systems. *Journal of the Association for Computing Machinery*, 38(3):691–729, 1991.

[GSV1] O. Goldreich, A. Sahai, and S. Vadhan. Honest Verifier Statistical Zero-Knowledge Equals General Statistical Zero-Knowledge. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 399–408, 1998.

[GSV2] O. Goldreich, A. Sahai, and S. Vadhan. Can Statistical Zero-Knowledge be Made Non-Interactive?, or On the Relationship of SZK and NISZK. In *CRYPTO '99*, pages 467–484, 1999.

[GV] O. Goldreich and S. Vadhan. Comparing Entropies in Statistical Zero-Knowledge with Applications to the Structure of SZK. In *Proceedings of the Fourteenth Annual IEEE Conference on Computational Complexity*, pages 54–73, Atlanta, GA, May 1999.

[GMR] S. Goldwasser, S. Micali, and C. Rackoff. The Knowledge Complexity of Interactive Proof Systems. *SIAM Journal on Computing*, 18(1):186–208, February 1989.

[GS] S. Goldwasser and M. Sipser. Private Coins versus Public Coins in Interactive Proof Systems. In S. Micali, editor, *Advances in Computing Research*, volume 5, pages 73–90. JAC Press, Inc., 1989.

[GB] D. Gutfreund and M. Ben-Or. Increasing the Power of the Dealer in Non-interactive Zero-Knowledge Proof Systems. In *ASIACRYPT '00: Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security*, pages 429–442, London, UK, 2000. Springer-Verlag. Journal version appeared as [BG].

[HILL] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396 (electronic), 1999.

[HR] T. Holenstein and R. Renner. One-Way Secret-Key Agreement and Applications to Circuit Polarization and Immunization of Public-Key Encryption. In *Advances in Cryptology CRYPTO 2005*, pages 478–493, New York, NY, USA, 2005. ACM Press.

[ILL] R. Impagliazzo, L. A. Levin, and M. Luby. Pseudo-random Generation from one-way functions (Extended Abstracts). pages 12–24.

[IY] R. Impagliazzo and M. Yung. Direct Minimum-Knowledge Computations (Extended Abstract). In *CRYPTO '87: A Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology*, pages 40–51, London, UK, 1988. Springer-Verlag.

[IOS] T. Itoh, Y. Ohta, and H. Shizuya. A language-dependent cryptographic primitive. *Journal of Cryptology*, 10(1):37–49, 1997.

[KW] A. Kitaev and J. Watrous. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In *Proceedings of the 32nd ACM Symposium on Theory of computing*, pages 608–617, 2000.

[Kob] H. Kobayashi. Non-interactive quantum perfect and statistical zero-knowledge. *ISAAC '03: International Symposium on Algorithms And Computation*, 2906:178–188, 2003.

[LFKN] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic Methods for Interactive Proof Systems. *Journal of the ACM*, 39(4):859–868, Oct. 1992.

[MV] D. Micciancio and S. Vadhan. Statistical Zero-Knowledge Proofs with Efficient Provers: Lattice Problems and More. In *CRYPTO*, pages 282–298, 2003.

[Nao] M. Naor. Bit Commitment Using Pseudorandomness. *Journal of Cryptology*, 4(2):151–158, 1991.

[NV] M.-H. Nguyen and S. Vadhan. Zero knowledge with efficient provers. In *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 287–295, New York, NY, USA, 2006. ACM Press.

[Oka] T. Okamoto. On Relationships Between Statistical Zero-Knowledge Proofs. *Journal of Computer and System Sciences*, 60(1):47–108, February 2000.

[OV1] S. J. Ong and S. Vadhan. Zero Knowledge and Soundness are Symmetric. In *EUROCRYPT '07: 26th Annual Conference on the Theory and Applications of Cryptographic Techniques*, 2007.

[OV2] S. J. Ong and S. Vadhan. An Equivalence between Zero Knowledge and Commitments, 2008. These proceedings.

[Pas] R. Pass and abhi shelat. Unconditional Characterizations of Non-Interactive Zero-Knowledge. In *CRYPTO '05*, pages 118–134. Springer Berlin / Heidelberg, 2005.

[PT] E. Petrank and G. Tardos. On the Knowledge Complexity of $\mathcal{NP}$. In *IEEE Symposium on Foundations of Computer Science*, pages 494–503, 1996.

[SV1] A. Sahai and S. Vadhan. Manipulating Statistical Difference. In P. Pardalos, S. Rajasekaran, and J. Rolim, editors, *Randomization Methods in Algorithm Design (DIMACS Workshop, December 1997)*, volume 43 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 251–270. American Mathematical Society, 1999.

[SV2] A. Sahai and S. Vadhan. A complete problem for statistical zero knowledge. *Journal of the ACM*, 50(2):196–249, March 2003.

[Sha] A. Shamir. IP = PSPACE. *Journal of the ACM*, 39(4):869–877, Oct. 1992.

[Vad] S. Vadhan. An Unconditional Study of Computational Zero Knowledge. *SIAM Journal on Computing*, 36(4):1160–1214, 2006. Special Issue on Randomness and Complexity.

[Wat1] J. Watrous. Limits on the Power of Quantum Statistical Zero-Knowledge. In *FOCS '02: Proceedings of the 43rd Symposium on Foundations of Computer Science*, pages 459–468, Washington, DC, USA, 2002. IEEE Computer Society.

[Wat2] J. Watrous. Zero-knowledge against quantum attacks. In *STOC '06: Proceedings of the thirty-eighth annual ACM Symposium on Theory of Computing*, pages 296–305, New York, NY, USA, 2006. ACM Press.